THE No 1 UK MAGAZINE FOR ELECTRONICS TECHNOLOGY & COMPUTER PROJECTS

EPE EVERYDAY PRACTICAL ELECTRONICS

www.epemag.com

LED MUSICOLOUR

Individual control of 16 strings of LEDs
16-bit DSP microcontroller
PWM LED drive
Operation via infrared remote

HIGH-TEMPERATURE THERMOMETER/THERMOSTAT

PRECISE MEASUREMENT AND RELAY CONTROL OF TEMPERATURE

WIN A MICROCHIP PIC32 GUI DEVELOPMENT BOARD

Teach-In 2014 Raspberry Pi – Part 1

EPE's comprehensive guide to Raspberry Pi

PLUS: **INGENUITY UNLIMITED, NET WORK, READOUT, CIRCUIT SURGERY, TECHNO TALK & INTERFACE**

OCT 2013 £4.40

10

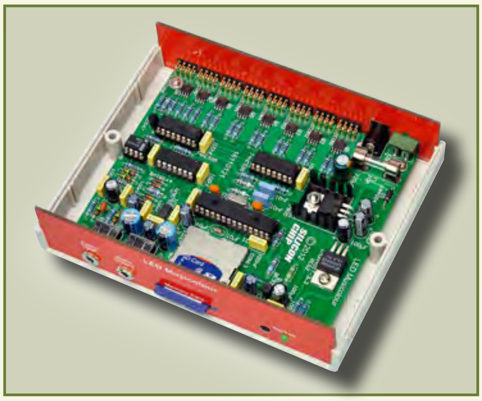
9 770262 361188

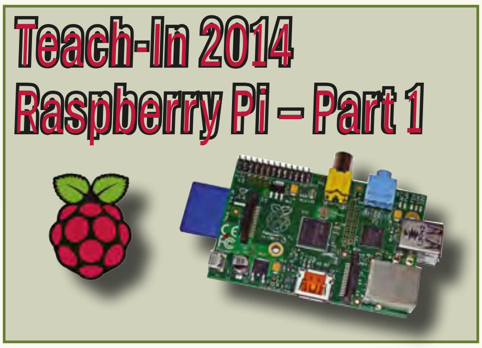
ISSN 0262 3617
• PROJECTS • THEORY •
• NEWS • COMMENT •
• POPULAR FEATURES •
VOL. 42. No 10 October 2013

EPE EVERYDAY PRACTICAL ELECTRONICS

INCORPORATING ELECTRONICS TODAY INTERNATIONAL

www.epemag.com





Projects and Circuits

Series and Features

Regulars and Services

Our November 2013 issue will be published on Thursday 3 October 2013, see page 72 for details.

QUASAR electronics
Solutions for Home, Education & Industry Since 1993

Quasar Electronics Limited
PO Box 6935, Bishops Stortford
CM23 4WP, United Kingdom
Tel: 01279 467799
Fax: 01279 267799
E-mail: sales@quasarelectronics.co.uk
Web: www.quasarelectronics.co.uk

All prices INCLUDE 20.0% VAT.
Postage & Packing Options (Up to 0.5Kg gross weight): **UK Standard 3-7 Day Delivery** - £3.95; **UK Mainland Next Day Delivery** - £8.95; **Europe (EU)** - £12.95; **Rest of World** - £14.95 (up to 0.5Kg).
!Order online for reduced price Postage (from just £1)
Payment: **We accept all major credit/debit cards. Make PO's payable to Quasar Electronics Limited.**
Please visit our online shop now for full details of over 1000 electronic kits, projects, modules and publications. Discounts for bulk quantities.

01279 Credit Card Sales 467799

PIC & ATMEL Programmers

We have a wide range of low cost PIC and ATMEL Programmers. Complete range and documentation available from our web site.

Programmer Accessories:
40-pin Wide ZIF socket (ZIF40W) £14.95
18Vdc Power supply (PSU121) £22.95
Leads: Parallel (LDC136) £3.95 / Serial (LDC441) £3.95 / USB (LDC644) £2.95

USB & Serial Port PIC Programmer

USB/Serial connection. Header cable for ICSP. Free Windows XP software. See website for PICs supported. ZIF Socket and USB lead extra. 18Vdc.
Kit Order Code: 3149EKT - **£49.95**
Assembled Order Code: AS3149E - **£64.95**
Assembled with ZIF socket Order Code: AS3149EZIF - **£74.95**

USB Flash PIC Programmer

USB PIC programmer for a wide range of Flash devices—see website for details. Free Windows Software. ZIF Socket and USB lead not included. Powered via USB port - no external power supply required.
Assembled with ZIF socket Order Code: AS3150ZIF - **£64.95**

ATMEL 89xxxx Programmer

Uses serial port and any standard terminal comms program. 4 LED's display the status. ZIF sockets not included. Supply: 16Vdc.
Kit Order Code: 3123KT - **£28.95**
Assembled Order Code: AS3123 - **£39.95**

Introduction to PIC Programming

Go from complete beginner to burning a PIC and writing code in no time! Includes 49 page step-by-step PDF Tutorial Manual, Programming Hardware (with LED test section), Win 3.11—XP Programming Software (Program, Read, Verify & Erase), and 1rewritable PIC16F84A that you can use with different code (4 detailed examples provided for you to learn from). PC parallel port.
Kit Order Code: 3081KT - **£16.95**
Assembled Order Code: AS3081 - **£24.95**

PIC Programmer Board

Low cost PIC programmer board supporting a wide range of Microchip® PIC™ microcontrollers. Requires PC serial port. Windows interface supplied.
Kit Order Code: K8076KT - **£34.95**

PIC Programmer & Experimenter Board

The PIC Programmer & Experimenter Board with test buttons and LED indicators to carry out educational experiments, such as the supplied programming examples. Includes a 16F627 Flash Microcontroller that can be reprogrammed up to 1000 times for experimenting at will. Software to compile and program your source code is included.
Kit Order Code: K8048KT - **£34.95**
Assembled Order Code: VM111 - **£44.95**

Controllers & Loggers

Here are just a few of the controller and data acquisition and control units we have. See website for full details. 12Vdc PSU for all units: Order Code 660.446UK £10.95

USB Experiment Interface Board

5 digital input channels and 8 digital output channels plus two analogue inputs and two analogue outputs with 8 bit resolution.
Kit Order Code: K8055NKT - **£27.95**
Assembled Order Code: VM110N - **£40.95**

Rolling Code 4-Channel UHF Remote

State-of-the-Art. High security. 4 channels. Momentary or latching relay output. Range up to 40m. Up to 15 Tx's can be learnt by one Rx (kit includes one Tx but more available separately). 4 indicator LED 's. Rx: PCB 77x85mm, 12Vdc/6mA (standby). Two *& Ten Channel versions also available.*
Kit Order Code: 3180KT - **£54.95**
Assembled Order Code: AS3180 - **£64.95**

Computer Temperature Data Logger

Serial port 4-channel temperature logger. °C or °F. Continuously logs up to 4 separate sensors located 200m+ from board. Wide range of free software applications for storing/using data. PCB just 45x45mm. Powered by PC. Includes one DS1820 sensor.
Kit Order Code: 3145KT - **£19.95**
Assembled Order Code: AS3145 - **£26.95**
Additional DS1820 Sensors - **£4.95 each**

Remote Control Via GSM Mobile Phone

Place next to a mobile phone (not included). Allows toggle or auto-timer control of 3A mains rated output relay from any location with GSM coverage.
Kit Order Code: MK160KT - **£10.72**

Most items are available in kit form (KT suffix) or pre-assembled and ready for use (AS prefix).

4-Ch DTMF Telephone Relay Switcher

Call your phone number using a DTMF phone from anywhere in the world and remotely turn on/off any of the 4 relays as desired. User settable Security Password, Anti-Tamper, Rings to Answer, Auto Hang-up and Lockout. Includes plastic case. 130 x 110 x 30mm. Power: 12Vdc.
Kit Order Code: 3140KT - **£79.95**
Assembled Order Code: AS3140 - **£94.95**

8-Ch Serial Port Isolated I/O Relay Module

Computer controlled 8 channel relay board. 5A mains rated relay outputs and 4 opto-isolated digital inputs (for monitoring switch states, etc). Useful in a variety of control and sensing applications. Programmed via serial port (use our new Windows interface, terminal emulator or batch files). Serial cable can be up to 35m long. Includes plastic case 130x100x30mm. Power: 12Vdc/500mA.
Kit Order Code: 3108KT - **£74.95**
Assembled Order Code: AS3108 - **£89.95**

Infrared RC 12–Channel Relay Board

Control 12 onboard relays with included infrared remote control unit. Toggle or momentary. 15m+ range. 112 x 122mm. Supply: 12Vdc/0.5A
Kit Order Code: 3142KT - **£64.95**
Assembled Order Code: AS3142 - **£74.95**

Audio DTMF Decoder and Display

Detect DTMF tones from tape recorders, receivers, two-way radios, etc using the built-in mic *or* direct from the phone line. Characters are displayed on a 16 character display as they are received and up to 32 numbers can be displayed by scrolling the display. All data written to the LCD is also sent to a serial output for connection to a computer. Supply: 9-12V DC (Order Code PSU303). Main PCB: 55x95mm.
Kit Order Code: 3153KT - **£37.95**
Assembled Order Code: AS3153 - **£49.95**

3x5Amp RGB LED Controller with RS232

3 independent high power channels. Preprogrammed or user-editable light sequences. Standalone option and 2-wire serial interface for microcontroller or PC communication with simple command set. Suitable for common anode RGB LED strips, LEDs and incandescent bulbs. 56 x 39 x 20mm. 12A total max. Supply: 12Vdc.
Kit Order Code: 8191KT - **£29.95**
Assembled Order Code: AS8191 - **£39.95**

FEATURED KITS in Everyday Practical Electronics

OCTOBER 2013

Everyday Practical Electronics Magazine has been publishing a series of popular kits by the acclaimed Silicon Chip Magazine Australia. These projects are 'bullet proof' and already tested Down Under. All Jaycar kits are supplied with specified board components, quality fibreglass tinned PCBs and have clear English instructions. Watch this space for future featured kits.

Theremin Synthesiser Kit MkII

Featured in EPE March 2011

Create your own eerie science fiction sound effects by simply moving your hand near the antenna. Easy to set up and build. Complete kit contains PCB with overlay, pre-machined case and all specified components.

- PCB: 85 x 145mm

KC-5475

POPULAR KIT!

£27.25*

Speed Control Kit for Induction Motors

£90.50*

Featured in EPE April 2013

Control induction motors* up to 1.5kW (2HP) to run machinery at different speeds or controlling a pool pump to save money. Also works with 3-phase motors. Full form kit includes case, PCB, heatsink, cooling fan, hardware and electronics (including revisions from the August Silicon Chip article). Please note that this is an advanced project for an experienced constructor.

KC-5509

*Does not work for motors with centrifugal switch

The mains power cord will need to be replaced with UK type.

Soft Start Kit for Power Tools

Featured in EPE July/August 2013

Stops that dangerous kick-back when you first power up an electric saw, router or other mains-powered hand tool. This helps prevent damage to the job or yourself when kick-back torque jerks the power tool out of your hand. Kit supplied with PCB, silk screened case, 2m power cord and specified electronic components.

- 240VAC 10A
- PCB: 81 x 59mm

KC-5511

£18.25*

The mains power cord will need to be replaced with UK type.

12/24VDC 20A Motor Speed Controller Kit

Featured in EPE March 2012

Control the speed of 12 or 24VDC motors from zero to full power, up to 20A. Features optional soft start, adjustable pulse frequency to reduce motor noise, and low battery protection. The speed is set using the onboard trimpot, or by using an external potentiometer (available separately, use **RP-3510 £0.77**).

- Kit supplied with PCB and all onboard electronic components
- Suitable enclosure UB3 case **HB-6013 £1.50** sold separately

KC-5502

£14.50*

Crazy Cricket & Freaky Frog Kit

Featured in EPE August 2013

A fun first project for a budding electronics enthusiast. Designed to imitate the chirping noise of a cricket or gentle croaking of a frog (alternates at power up), while keeping its location secret to annoy other family members. It activates in darkness and stops when disturbed by light. Kit supplied with PCB, pre-programmed IC, battery and electronic components.

- PCB: 30 x 65mm

KC-5510

£7.25*

Low Capacitance Adaptor for DMM Kit

Featured in EPE August 2013

This kit is a nifty little adaptor that allows a standard digital multimeter to measure very low values of capacitance from less than one picofarad to over 10nF. It will allow you to measure tiny capacitors or stray capacitances in switches, connectors and wiring. Kit comes complete with PCB, components and case.

- Requires 9V battery

KC-5493

£12.75*

FEATURED THIS MONTH

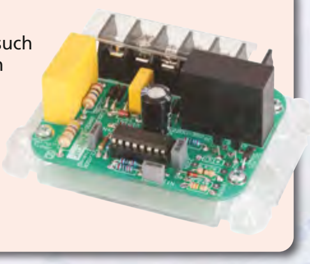
Mains Timer Kit for Fans and Lights

This simple circuit provides a turn-off delay for a 230VAC light or a fan, such as a bathroom fan set to run for a short period after the switch has been turned off. The circuit consumes no stand by power when load is off. Kit supplied with PCB, case and electronic components. Includes 100nF capacitor for 1 min to 20 mins. See website for a list of alternate capacitors for different time periods between 5 seconds to 1 hour.

- Handles loads up to 5A
- PCB: 60 x 76mm

KC-5512

£14.50*



ATTENTION KIT BUILDERS

Can't find the kit you are looking for? Try the Jaycar Kit Back Catalogue

Our central warehouse keeps a quantity of older and slow-moving kits that can no longer be held in stores. A list of kits can be found on our website. Just go to jaycar.co.uk/kitbackcatalogue

Ultrasonic Antifouling Kit for Boats

Now includes pre-built transducer at no extra cost!

Featured in EPE September/October 2012

Marine growth electronic antifouling systems can cost thousands. This project uses the same ultrasonic waveforms and virtually identical ultrasonic transducers mounted in a sturdy polyurethane housings. By building it yourself you save a fortune! Standard unit consists of control electronic kit and case, pre-built ultrasonic transducer and gluing components and housings. The single transducer design of this kit is suitable for boats up to 10m (32ft; boats longer than about 14m will need two transducers and drivers). Basically all parts supplied in the project kit including wiring.

- 12VDC
- Suitable for power or sail
- Can be powered by a solar panel/ wind generator
- PCB: 104 x 78mm

KC-5498

£90.50*

POPULAR KIT!

Also available Pre-built:

Dual output, suitable for vessels up to 14m (45ft) **YS-5600 £309.25***

Quad output, suitable for vessels up to 20m (65ft) **YS-5602 £412.25***

For more details on each kit visit our website www.jaycar.co.uk

Jaycar Electronics
Better, More Technical

FREE CALL ORDERS: 0800 032 7241

VOL. 42 No. 10 OCTOBER 2013

Editorial Offices:
EVERYDAY PRACTICAL ELECTRONICS EDITORIAL Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU
Phone: (01202) 880299. **Fax:** (01202) 843233.
Email: fay.kearn@wimborne.co.uk
Website: www.epemag.com
See notes on **Readers' Technical Enquiries** below – we regret technical enquiries cannot be answered over the telephone.

Advertisement Offices:
Everyday Practical Electronics Advertisements
113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU
Phone: 01202 880299 **Fax:** 01202 843233
Email: stewart.kearn@wimborne.co.uk

Editor:	MATT PULZER
Subscriptions:	MARILYN GOLDBERG
General Manager:	FAY KEARN
Graphic Design:	RYAN HAWKINS
Editorial/Admin:	01202 880299
Advertising and Business Manager:	STEWART KEARN 01202 880299
On-line Editor:	ALAN WINSTANLEY
Publisher:	MIKE KENWARD

EVERYDAY PRACTICAL ELECTRONICS

Teach-In 2014

Eighteen months ago, I wrote an enthusiastic *Editorial* about a promising new British electronic innovation called Raspberry Pi. It was described as a 'barebones computer on a credit-card-sized PCB' at the remarkable price of just £15 (now a little more). With its USB port for mouse/keyboard and an HDMI interface for plugging into a TV for display purposes, the idea was to give anyone, but especially interested children, the opportunity to play with and program a simple computer without worrying about cost, complex operating systems or all the other complications that come with using a conventional Windows or Apple OS X computer.

It was a great idea, and the initial production run of 10,000 seemed like a reasonable gamble. In fact, it was an extraordinary under-estimation. Come January 2013, a million had been sold round the world, and by now, production must have hit 1.5 million.

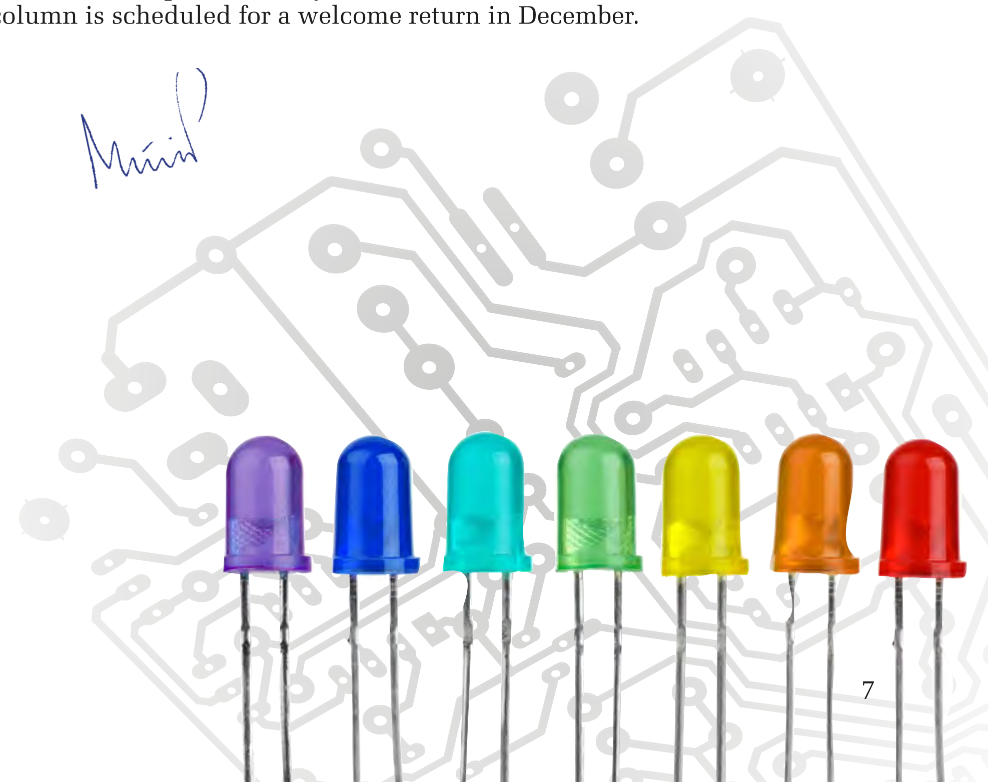
Perhaps the best measure of its success is the hundreds of supporting devices and millions of lines of code that have been generated. Software and hardware galore are available for Pi experimenters.

We know that *EPE* readers are keen to get started with Pi, so I am genuinely excited by this month's start of our *Teach-In 2014* series, which will be devoted solely to Raspberry Pi. In Part 1, *Teach-In* authors Mike and Richard Tooley have kicked off with a mouthwatering introduction and a glimpse of the slices of Pi to look forward to in future issues. Software, hardware, interfacing, construction techniques and projects are all lined up for your electronic delight.

We know you will enjoy this exciting series, and as always, we welcome feedback, questions and your moments of triumph as you succeed in creating new applications for this wonderfully versatile building block. To that end, we have set aside a dedicated section in *EPE Chat Zone* for you to share frustration, success and tips.

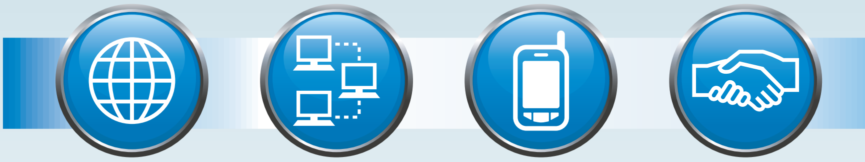
No PIC user left behind

We know our dedicated band of PIC users is feeling a little forlorn at the moment – Mike Hibbet, our resident PIC guru, has had an exceptionally busy time in his day job, and needed to take a few months off *EPE*. However, I am pleased to say that he will soon be back, and his *PIC n' Mix* column is scheduled for a welcome return in December.



NEWS

A roundup of the latest Everyday News from the world of electronics



Satnav companies search for route to success – by Barry Fox

The standalone satnav market is in a state of confusing upheaval. The two main vendors, Garmin and TomTom, must now compete with the GPS map navigation that comes free or cheap with virtually all smartphones – as well as with each other. Both companies now have to sacrifice revenue from map update subscriptions and are offering free updates for the lifetime of some devices.

A falling market

Garmin estimates that the total market for car satnavs is now 10% down to £170 million a year and reliant mainly on replacements.

In an effort to win share in the replacement market, Garmin is also claiming exclusivity on a new jam-busting 'Digital Traffic' system that relies on DAB data, and is free to use for the device lifetime. 'Real Directions', which give instructions like 'turn left at the church', is also claimed to be 'Exclusive to Garmin'. TomTom promises images of buildings and landmarks in 3D.

Garmin recently held a press and trade briefing at the London Transport Museum in London, to try and inject some clarity.

Broadly speaking, most new Garmin devices costing over £100 update their maps four times a year when connected to a PC.

DAB devices

Garmin devices costing over £150 also come with free 'Digital Traffic'. The satnav has a built-in DAB chip, which continually receives TPEG (Transport Protocol Experts Group) data on traffic flow. The data comes from US company Inrix, which monitors road sensors, cameras and cellphone networks, calculates average speeds of road traffic and warns of jams. If DAB reception fails, the satnav switches to receiving slower Traffic Message Channel data by FM. The user pays nothing for data reception because no cellphone connection is needed.

Smartphones offer more and more sophisticated navigational capability for free, forcing traditional satnav vendors such as Garmin and TomTom to revisit their business model

TomTom has been far less proactive in briefing, but says its new Go, Via and Start series also come with free lifetime maps, with four updates a year. However, Halfords has been offering a 'Special Promotion' which bundles 'Free Lifetime Maps Worth £79.99' with selected TomTom satnavs, see: **http://tinyurl.com/kr9dufm**

Cellphone operation

TomTom is sticking with its existing Traffic system, which logs traffic in the same way as Garmin but communicates by cellphone connection, either via a smartphone or built-in cellular chips. 'Extra costs may apply' warns TomTom.

TomTom's spokeswoman explains: 'An 'always connected' device has a SIM and therefore incurs no data charges. A 'smartphone connected' device uses the SIM in your mobile… we'll make this clear on a dedicated website, with approximations of data charges.'

Speed camera charges

Both companies are still clinging to one revenue-earning option, map updates on the location of speed cameras. After short free trials, TomTom and Garmin both charge for updates. (However, holidaymakers to France should note that it is illegal to use any kind of speed camera detection or warning, so the feature must be disabled from all satnavs used there.)

BBC/YouView Red Button confusion

The last minute compromise decision by the BBC to broadcast the Wimbledon tennis finals in 3D using the UK's Red Button interactive system, rather than on a regular HD DTTV channel as previously, has exposed a practical problem with the YouView personal video recorders (PVRs) which the BBC helped design. Even though the Red Button channel displays on screen, the YouView box usually records something different.

Initially, YouView's customer support service was unaware of the issue, but the company now says: 'its technical team is now looking into changing the box design or displaying warnings on screen'.

YouView is a partnership between the UK's main telecommunications operators and public service broadcasters (BBC, ITV, Channel 4 and Channel 5). The YouView set-top box is a free-to-air hybrid device, which receives and records terrestrial HDTV and displays – but does

not record – past programmes using a broadband connection.

The YouView box also supports the UK's Red Button system, which lets broadcasters put a red dot at the top right of the screen to tell the listener that extra content is available by pressing a red button on the TV remote. The TV then temporarily retunes to a TV sub-channel, buried deep in the electronic programme guide (EPG).

Until recently, broadcasters used the Red Button only for extra news or weather information, but the BBC is now using the Red Button to show limited interest video streams, such as highlights of music festivals or second-stage acts – and now 3D, which has very limited viewer appeal.

When used to watch Wimbledon, a YouView box successfully displayed the selected match in conventional side-by-side 3D when the Red Button was pressed. But when the box was then switched to record the 3D on screen it recorded only a 2D image.

At first, YouView's customer support insisted that wrong buttons must have been pressed. But now both the BBC and YouView's design team have confirmed that, 'it is a function of the box' that even when YouView is displaying a secondary video stream, such as 3D, it will record the main stream, in 2D.

YouView says it now hopes to modify the box design, hopefully with a software upgrade. Until then, viewers who want to record Red Button 3D or second-stage bands will have to avoid the intuitive approach, become tech-savvy and dig deep into the EPG to find the secondary stream by channel number, and then record that.

3.2-inch LCD touch screen

Parallax has launched an intelligent LCD touch screen display module to add interactive, multimedia functionality to microcontroller projects. The module (uLCD-32PTU) is from 4D Systems, a leading developer of LCD screen technology. Implementation of the screen into projects is made easy with with 4D Systems' comprehensive 'Workshop4' IDE tool suite. The tool helps set up graphical interfaces that can be used with a simple serial connection to microcontrollers. Key features include: 240 × 320 resolution and 65K true-to-life colours; a microSD card slot and 14KB of flash memory for storing code.

More details at: **www.parallax.com** (Item code 28083).

3D printing with metals

Researchers from North Carolina State University have developed 3D printing technology and techniques to create free-standing structures made of liquid metal at room temperature.

'It's difficult to create structures out of liquids, because liquids want to bead up. But we've found that a liquid metal alloy of gallium and indium reacts to the oxygen in the air at room temperature to form a 'skin' that allows the liquid metal structures to retain their shapes,' says Dr Michael Dickey, an assistant professor of chemical and biomolecular engineering at NC State and co-author of a paper describing the work.

The researchers developed multiple techniques for creating these structures, which can be used to connect electronic components in three dimensions. While it is relatively straightforward to pattern the metal 'in plane' (all on the same level) these liquid metal structures can also form shapes that reach up or down.

One technique involves stacking droplets of liquid metal on top of each other, much like a stack of oranges at the supermarket. The droplets adhere to one another, but retain their shape – they do not merge into a single, larger droplet.

3D printing with metal – exactly what electronics hobbyists have been waiting for!

Another technique injects liquid metal into a polymer template, so that the metal takes on a specific shape. The template is then dissolved, leaving the bare, liquid metal in the desired shape. The researchers also developed techniques for creating liquid metal wires, which retain their shape even when held perpendicular to the substrate.

Dickey's team is currently exploring how to further develop these techniques, as well as how to use them in various electronics applications and in conjunction with established 3D printing technologies, which are essentially plastic-based.

A video of the process is available at: **http://youtu.be/ql3pXn8-sHA**

Silver fibres for wearable electronics

Scientists at the UK's National Physical Laboratory (NPL) have developed a way to print silver directly onto fibres. This technique could make integrating electronics into all types of clothing simple and practical. This has many potential applications in sports, medicine, consumer electronics and fashion.

Most current plans for wearable electronics require weaving conductive materials into fabrics, which offer limited flexibility and can only be achieved when integrated into the design of the clothing from the start. NPL's technique could allow lightweight circuits to be printed directly onto complete garments.

The technique involves chemically bonding a nano-silver layer onto individual fibres to a thickness of 20nm. The silver layer fully encapsulates fibres and has good adhesion and excellent conductivity.

Supertoy – natural talking teddy bear launched

A robotic toy that can talk naturally has been launched. Supertoy is a talking teddy bear with a mind of its own and the ability to hold real conversations. Its creators, Supertoy Robotics, are on Kickstarter, aiming to secure funds to put the electronically advanced toy into production.

The robot bear mimics awareness and has his own autonomous thoughts. The software copies real human speech patterns, and the bear uses smartphone technology as its 'brawn'. To enhance human-like speech, Supertoy's robotic mouth moves in synch' with what it says and inbuilt speakers enhance the volume of his voice. Users download a free app and zip a smartphone into the bear's back to give the teddy its unique personality and ability to converse. Internet access to the bear's server 'brain' allows it to react realistically to things that children say; a sort of 'Siri for children'. More details at: **www.kickstarter.com**

Light up your music with the . . .

LED MUSICOLOUR

HALLOWEEN, BONFIRE NIGHT and Christmas are creeping up on us. If you don't have a light show up and running already you'd better get started! Our new *LED Musicolour* makes it easier than ever. It drives up to 16 sets of LEDs directly. These can be strips, strings, single LEDs or a set – whatever, as long as they run off 12-24V DC.

You can even build multiple *LED Musicolours* and run them in parallel, to control 32 or even 64 sets of LEDs.

You can drive the *LED Musicolours* using any line source, such as CD or MP3 player, or you plug in an SD card which has been loaded with WAV files. In the latter case, it can be a self-contained sound and light controller with no need for any extra hardware apart from a power supply.

The unit supports high-capacity SDHC cards, so you can load it up with lots of music (organised in folders) and use a universal infrared remote control to skip through them. If you build more than one, you can use one as the 'master' to play the audio and feed it to the others for a synchronised light show, as well as to an amplifier so you can hear the music at the same time.

The *LED Musicolours* uses a 40MHz, 16-bit digital signal controller which is actually a specialised DSP (digital signal processor) microcontroller. It is powerful enough to do real-time frequency analysis using a Discrete Fourier Transform. The unit also incorporates a Wolfson WM8759 audio DAC for

Part 1: By NICHOLAS VINEN

Now you can have a kaleidoscope of colour which continually changes in time to music. This consists of 16 strings of LEDs that are individually controlled by 16 frequency bands. Louder signals in each of those bands means that the respective LED string will be brighter. Use it for a Christmas light show, a disco or just for fun when playing music.

good-quality line level sound output. It all fits into a tiny plastic case, which seems quite innocuous considering all the fancy processing it is performing.

Echoes of the past

This unit is intended as an easier-to-use version of the *DSP Musicolour* which was published in *EPE* from May-August 2010. It was also somewhat inspired by the *Digital Lighting Controller* featured in the October-December 2012 issues.

The *Digital Lighting Controller* controls the brightness of up to 32 mains-powered lights or LED strips, in time to music. But its light sequences are pre-arranged, ie, you program a specific sequence to go along with each sound file. That is a somewhat laborious process, but it gives you full control over the light show. It has no option to feed in external audio and its output sound quality is a bit so-so.

Also, the *Digital Lighting Controller* required you to build a master unit and between one and four slave units, with one slave controlling eight lights. The new *LED Musicolour*, on the other hand, is fully self-contained and can control 16 LED strips per unit. And as explained earlier, you can chain multiple *LED Musicolour* units together if you need to control more LEDs. So to sum up, the *LED Musicolour* is more flexible (having an audio input) and is easier to build and set up, but doesn't give you quite as much control as the *Digital Lighting Controller* from

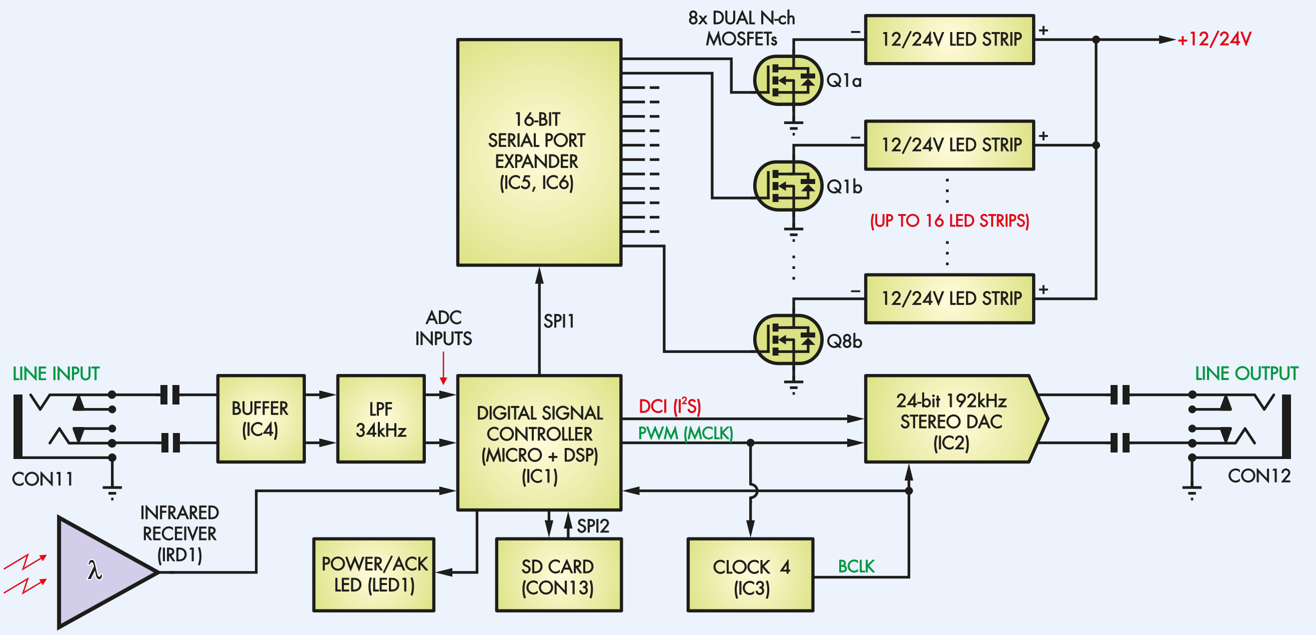


Fig.1: block diagram for the *LED Musicolour*. At its heart is IC1, a dsPIC33 digital signal controller/microcontroller. IC5 and IC6 are used to drive the MOSFETs which control the brightness of up to 16 LED strips. Audio can either be fed in to CON11 or played back from an SD card in CON13, via stereo DAC IC2, to CON12

2012. Having said that, the *LED Musicolour's* light shows are quite impressive and it is much simpler to build.

How it works

Take a look at the block diagram of Fig.1. At its heart is IC1, a dsPIC33 digital signal controller (DSC). As stated above, this is a 40MHz 16-bit microcontroller and it features a fixed-point DSP unit and extra features to enhance performance, such as multi-ported random access memory (RAM).

The brightness of the 16 strips of LEDs is controlled by switching 16 MOSFETs at 200Hz, using pulse-width modulation (PWM). IC1 only has 28 pins and that isn't enough to drive the MOSFETs directly and leave some over for other purposes. So the MOSFET gates are driven from the outputs of two 8-bit serial-to-parallel latch ICs (IC5 and IC6) which act as a port expander. IC1 updates their output state using one of its internal serial peripheral interface units, SPI1.

When a given output from IC5 or IC6 is high, this turns on the corresponding MOSFET, which sinks current from the negative supply line of the corresponding LED strip. The positive supply lines are permanently tied to the 12-24V supply. By controlling the proportion of the time that the MOSFET is on, we control the average current through the LED strip and therefore its brightness.

Audio can be fed into the unit via 3.5mm phono socket CON11. The audio signals (left and right) are then AC-coupled to IC4, which buffers them and applies a DC offset (~1.65V). This offset is necessary since the circuitry runs off a single DC supply rail.

Before being fed to IC1, the signals go through a low-pass filter with a corner frequency of 34kHz. This removes high-frequency signals that could cause aliasing when IC1 digitises the audio at a sampling rate of 48kHz, using its internal 12-bit analogue-to-digital converter.

Alternatively, the unit can play back the audio from an SD card at CON13. IC1 reads WAV audio data off the card using its other SPI peripheral, SPI2. It then simultaneously analyses the data to determine the brightness of the LED strips and sends it to IC2, a stereo digital-to-analogue converter (DAC), using its data converter interface (DCI) unit.

Audio data is transmitted from IC1 to IC2 in I^2S format. The DAC also requires a 'master clock' which is a multiple of the sample clock – here, ×192. For example, when playing 48kHz audio, the master clock is 9.216MHz. This is generated by IC1 using a PWM output, which outputs a rate proportional to its instruction clock.

The ratio we are using is 4:1, which gives an instruction clock of 34-37MHz, depending on the audio sampling rate. This is derived from an 8MHz crystal by changing the multiplication and division factors of IC1's internal phase-locked loop (PLL).

IC3 acts as a clock divider to convert the master clock (192 times sample rate) to the appropriate bit clock rate for the I^2S stream. This has a fixed ratio; we are transmitting 48 bits for each sample (24 per channel) which means we need a division ratio of 192 ÷ 48 = 4. This bit clock is fed to both IC1 and IC2. The audio from IC2 is AC-coupled to CON12, a 3.5mm phono socket.

If you want to control the *LED Musicolour* with an infrared remote control, which is handy when playing back WAV files (but not strictly necessary), the commands are received by IRD1 and sent to IC1 which decodes them. The Power/Ack LED (LED1) flashes in response; it is normally lit while the unit is powered, to indicate that it is operating.

Circuit description

Refer now to Fig.2 for the full circuit details. As shown, IC1 sends serial data to and controls shift registers IC5 and IC6 using four data lines: DS

LED Musicolour Parts List

1 PCB, available from the *EPE PCB Service,* code 16110121, 103mm × 118mm
1 front panel, 134.5mm × 30mm
1 rear panel, 134.5mm × 30mm
1 instrument case, 140mm × 110mm × 35mm
1-16 12V or 24V LED strips with 2-pin or 4-pin sockets
1 12V or 24V DC power supply sufficient for LED strips
1 SD or SDHC card (optional)
1 universal infrared remote control (optional)
2 PCB-mount M205 fuse clips (F1)
1 10A M205 fuse (F1)
8 8-pin dual row right-angle pin headers, 2.54mm pitch (may be snapped from larger headers) (CON1-CON8)
1 PCB-mount DC socket (CON9)
1 2-way right-angle pluggable terminal block (CON10)
2 PCB-mount switched 3.5mm phono sockets (CON11, CON12)
1 Oupin SMD SD card socket (or equivalent) (CON13)
1 8MHz HC-49 crystal (X1)
1 6073B type TO-220 heatsink
1 M3 × 6mm machine screw
1 M3 × 10mm machine screw
2 M3 shakeproof washers
2 M3 hex nuts
4 No.4 × 9mm self-tapping screws
1 28-pin narrow IC socket
1 8-pin IC socket (optional)
1 14-pin IC socket (optional)
2 16-pin IC sockets (optional)

Semiconductors
1 dsPIC33FJ128GP802-I/SP microcontroller programmed with 1611012A.hex (IC1)
1 WM8759 24-bit 192kHz stereo DAC (IC2) (Element14 1776274)
1 74HC393 dual binary counter (IC3)
1 LM358 dual op amp (IC4)
2 74HC595 serial-to-parallel shift registers (IC5, IC6)
8 Si4944DY dual SMD MOSFETs (or equivalent) (Q1-Q8)
1 BC547 *NPN* transistor (Q9)
1 BC327 *PNP* transistor (Q10)
1 7805T 5V 1A regulator (REG1)
1 LM3940IT-3.3 3.3V LDO regulator (REG2)
1 1N4004 1A diode (D1)
4 BAT85 small-signal Schottky diodes (D2-D5)
1 green 3mm LED (LED1)
1 3-pin infrared receiver (IRD1)

Capacitors
1 220µF 25V low-ESR electrolytic
2 220µF 16V electrolytic
5 100µF 25V electrolytic
4 10µF 16V electrolytic
1 10µF 6V SMD ceramic (3216)
13 100nF MKT or MMC
2 10nF MKT or MMC
2 100pF ceramic
2 33pF ceramic

Resistors (0.25W, 1%)

2 1MΩ	6 1kΩ
2 120kΩ	2 470Ω
3 100kΩ	1 220Ω
3 47kΩ	19 100Ω
5 10kΩ	1 10Ω
1 4.7kΩ	

(serial data), SRCK (serial clock), LCK (latch control) and MR (master reset). While microcontroller IC1 runs off 3.3V, IC5 and IC6 run off 5V, to drive MOSFETs Q1a-Q8b with sufficient voltage to switch them on.

Unfortunately, a 74HC595 with a 5V supply will not work reliably with 3.3V input signals (according to the data sheet), so these signals need to be level shifted to 5V. For SRCK, DS and LCK, the corresponding IC1 pins (RB8, RB6 and RB5) are set as open-drain outputs. These pins are 5V tolerant and with 1kΩ pull-up resistors to +5V, they can operate to at least 1MHz.

For MR, we have used a different arrangement because we want this line to be low by default, keeping the outputs of shift registers IC5 and IC6 off until IC1 brings it high. NPN transistor Q9 acts as an inverter/level shifter; the 100kΩ pull-up resistor between its base and collector holds it on when IC1 is not driving it.

IC5 and IC6 drive the MOSFET gates via 100Ω series resistors which form low-pass RC filters in conjunction with the MOSFET gate capacitances (about 1nF each). This prevents oscillation and overshoot when switching on or off, due to copper track inductance. We are using eight dual-MOSFETs to keep the cost and size down. Each can switch up to 9.4A at 30V.

The LED strings are connected to a series of 4-pin headers. Their outer two pins are connected to the high side of the supply (12-24V) and the inner two to the MOSFET drain. That way, you can plug the LED strip connector in either way around and it will still work. It also gives more contact area to safely allow up to 2A per LED string.

Audio inputs

Turning now to the audio inputs, the left and right channel signals from CON11 first pass through low-pass RC filters consisting of 100Ω resistors and 100pF capacitors. These filters prevent RF (radio frequency) signals from entering the device. There are also 1MΩ bias resistors in case the source's output is AC-coupled. The value of these resistors can be lowered if you want to feed in the output of an iPod, eg, to 1kΩ each.

The signals are then AC-coupled by 100nF capacitors to a resistive divider/DC bias network. This forms a high-pass filter with a –3dB point at 7Hz. It also sets the input impedance of the device to 1MΩ||220kΩ = 180kΩ.

The 100kΩ/120kΩ dividers allow an input signal of up to $2.2V_{RMS}$ before clipping. The signal fed into IC1 is limited by its supply rails to 3.3V peak-to-peak. This translates to 3.3V ÷ (2√2) = $1.17V_{RMS}$. However, many CD/DVD/Blu-ray players, computers and so on will put out $2V_{RMS}$ or more. So we attenuate the signal by a factor of 120kΩ ÷ (100kΩ + 120kΩ) = 0.55 to allow for this.

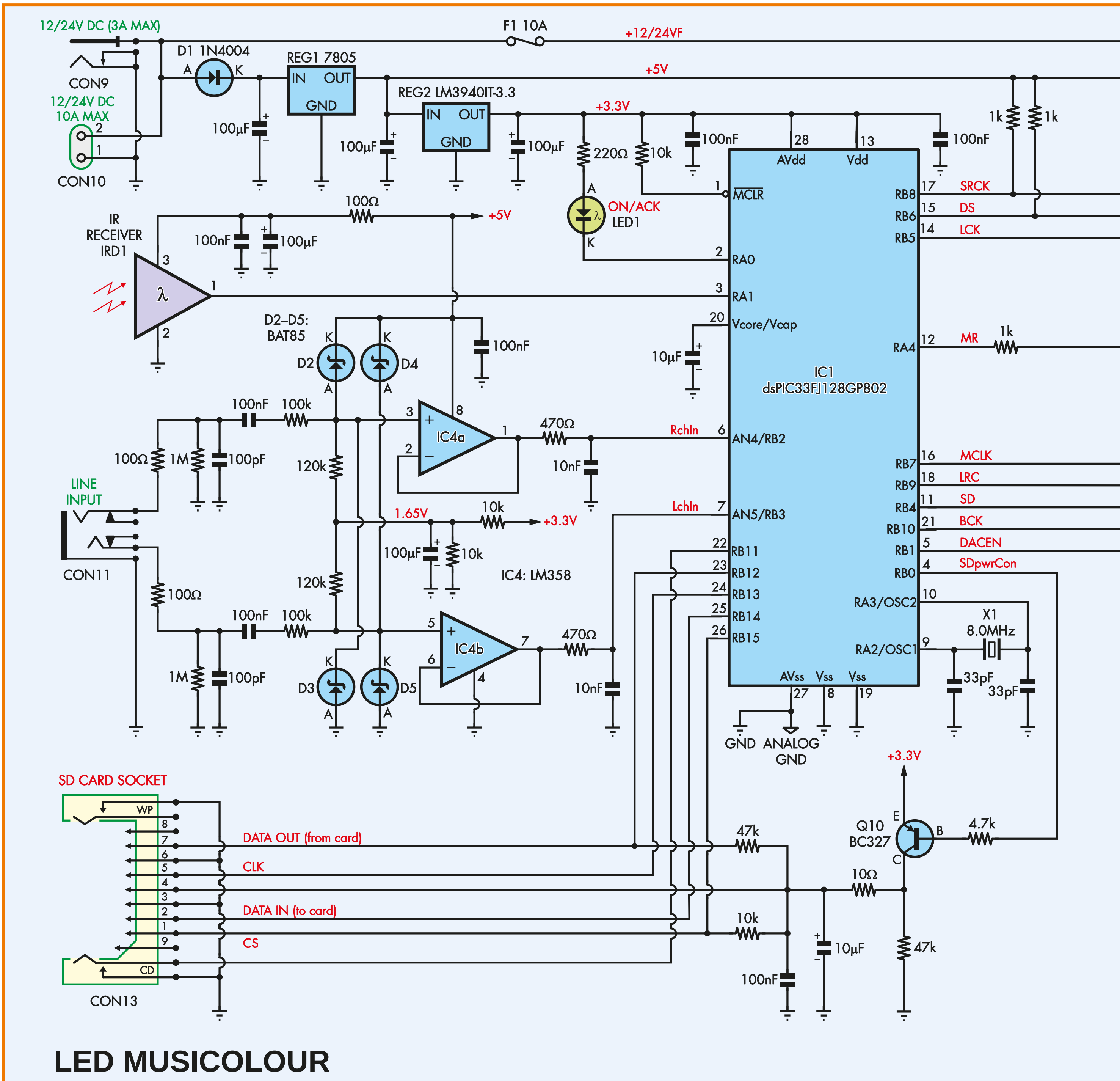
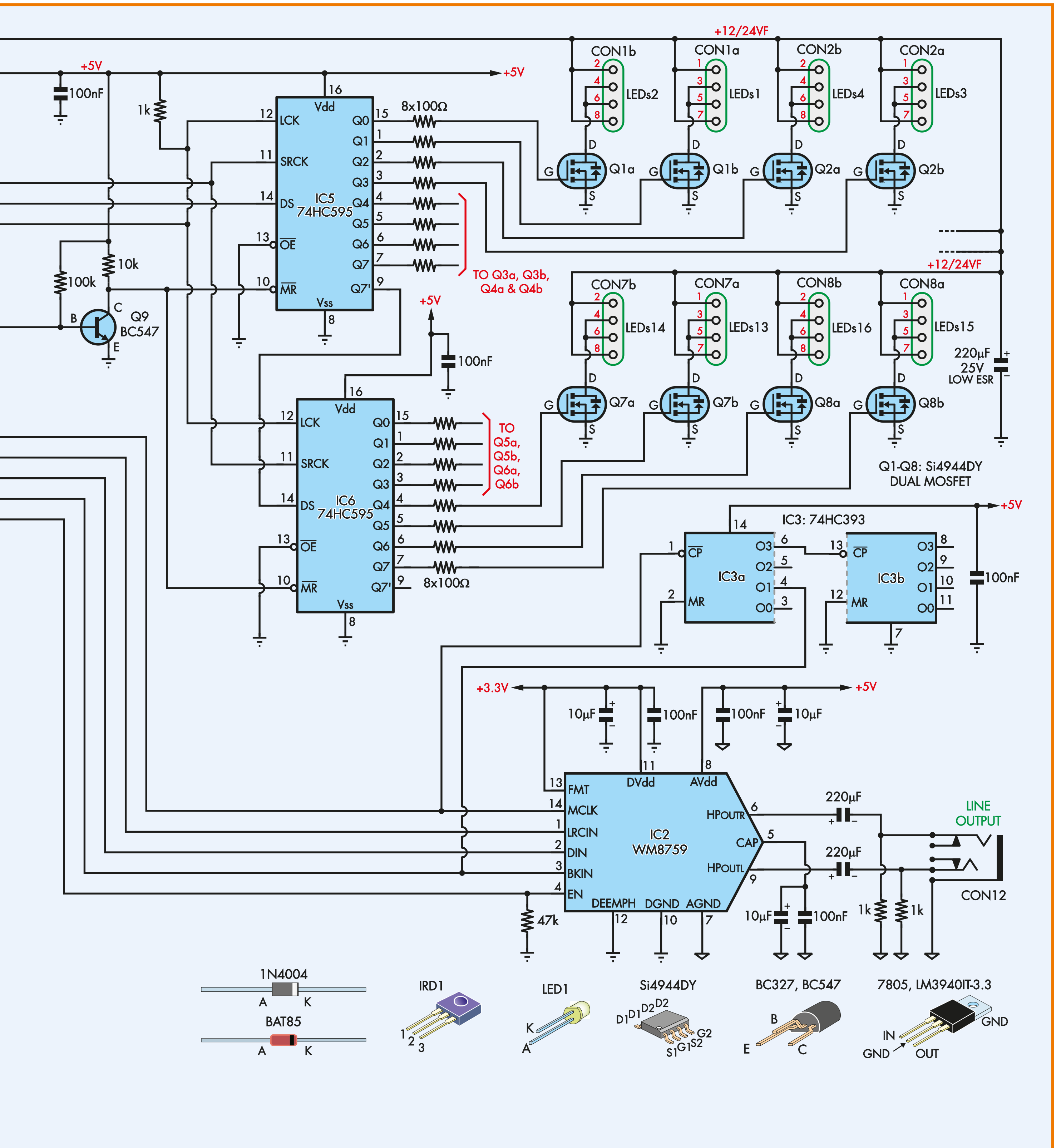


Fig.2: the *LED Musicolour* circuit diagram. Audio fed into CON11 is filtered, AC-coupled, attenuated, buffered and filtered again before passing to IC1's internal ADC. IC1 communicates with IC5 and IC6 via a serial (SPI) bus and these ICs drives MOSFETs Q1a-Q8b. Another SPI bus is used to read/write the SD card in CON13, while a similar I²S serial bus is used to send audio data to stereo DAC IC2. IC3 ensures that the DAC's master clock (MCLK) and serial data bit clock (BCK) are synchronised.

The bottom end of these resistive dividers is connected to a half-supply point of about 1.65V, derived using two 10kΩ resistors and filtered by a 100µF capacitor. The signals fed to analogue input pins AN4 and AN5 of IC1 thus swing symmetrically around its half-supply point. Op amps IC4a and IC4b buffer the input signal so that the following RC low-pass filter (470Ω/10nF) does not load up the signal source or previous stages.

Schottky diodes D2-D5 protect the inputs of IC4 from going outside its supply rails when the unit is not powered. While not strictly necessary, this makes the circuit more 'bulletproof'.

The low-pass RC filters following IC4a and IC4b both have a –3dB point of 34kHz and are tuned to the



sampling rate of IC1's ADC (48kHz). If there is much signal above the Nyquist frequency (48kHz/2 = 24kHz), this can cause aliasing, which will result in spurious sub-harmonics being detected. The filters (mostly) prevent that from happening.

Audio outputs

As explained earlier, the master clock for DAC IC2 (ie, from output RB7 of IC1) is generated by a PWM peripheral which divides IC1's instruction clock by four. This is fed to IC2 and also to IC3a's $\overline{CP}$ clock input (pin 1). Its O1 output is one-quarter of the input frequency and this is the audio data bit clock, BCK. BCK is fed to both IC1 and IC2, which use it to send and receive the audio data respectively.

IC3a's O3 output (1/16th MCLK) is connected to input $\overline{CP}$ of IC3b, the

Specifications

- **LED voltage:** 12-24V DC
- **LED current:** up to 10A total (ie, 120-240W maximum)
- **Number of LED strings:** up to 16 per unit
- **LED control method:** PWM, 200Hz, 255 brightness steps
- **LED connectors:** 2-pin or 4-pin male headers, 2.54mm pitch
- **Audio input:** 0.5-2.2V RMS nominal, 180kΩ || 100pF input impedance
- **Audio output:** 1.1V RMS, THD+N 0.004%, signal-to-noise ratio 100dB
- **Audio file support:** 8-48kHz 16-bit mono or stereo WAV files
- **Maximum directory depth:** eight levels
- **Maximum files per directory:** 100
- **Control method:** universal infra-red remote (optional)
- **Dimensions:** 140 x 110 x 35mm
- **Supply voltage:** 12-24V DC
- **Current drain:** ~110mA at 12V

other half of the dual binary counter. This is for testing purposes; the IC3b outputs give various frequencies related to the sampling rate being used. For example, with a 48kHz sampling rate, pin 9 of IC3 (O2) will measure 72kHz (48kHz × 1.5) and pin 8 (O3) will measure 36kHz (48kHz × 0.75).

The I^{2}S data stream from IC1's DCI peripheral comes from pins RB9 (LRC) and RB4 (SD). LRC is the left/right clock and represents the sampling rate (eg, 48kHz). It is produced by the data framing output of the DCI module. SD is the serial audio data and this is clocked according to the signal received at its RB10 input (pin 21).

By default, IC2 is in standby mode, as its enable pin (pin 4) is pulled to ground by a 47kΩ resistor. When IC1 is transmitting audio, it brings output RB1 (pin 5) high, pulling IC2's enable pin high and thus turning on the DAC.

IC2 has a pair of bypass capacitors (MKT and electrolytic) between each pair of supply pins, digital (DVDD/GND) and analogue (AVDD/AGND). Its format input (pin 13) is tied high to 3.3V, setting it to I^{2}S mode. Its DEEMPH input (pin 12) is low since we don't need digital de-emphasis. A pair of capacitors between its CAP pin (pin 5) and AGND filter its internal half-supply rail.

Audio is available at HPOUTL (pin 9) and HPOUTR (pin 6). These signals are AC-coupled using 220µF electrolytic capacitors, as IC2 can drive loads down to 16Ω. The 1kΩ DC bias resistors set the average output level to 0V.

SD card interface

The SD card interface is quite simple and is a tweaked version of the same interface we have used in the past. Microcontroller input RB11 (pin 22) is connected to the socket's Card Detect line, which is pulled to ground when a card is inserted. IC1's weak internal pull-up is enabled for RB11, allowing it to sense when this occurs.

The SD card is operated in 1-wire mode and IC1's SPI2 unit is used to send and receive data. This is mapped to pins RB15 (pin 26, card select), RB14 (pin 25, data to card), RB13 (pin 24, serial clock) and RB12 (pin 23, data from card). The card select (CS) and data from card (DATA OUT) lines are pulled up to V_{DD} to prevent any card operations from occurring when IC1 is reset or being programmed.

The SD card's V_{DD} line is not connected directly to 3.3V, but rather switched by PNP transistor Q10, which is controlled by output RB0 (pin 4) of the micro. This allows it to turn power on for the card only after it has been inserted.

The associated 100nF and 10µF capacitors bypass the SD card's supply, while a 10Ω series resistor prevents excessive current from being pulled from the 3.3V rail when the SD card is first powered up. A 47kΩ bleeder resistor shunts any leakage from Q10 so that the supply bypass capacitors don't charge up when it is off.

Remaining parts

Infrared receiver IRD1 detects IR pulses from the remote and converts them into digital signals, which it sends to input RA1 (pin 3) of IC1. IC1 then uses a pin-change interrupt handler routine to decode the Philips RC5-coded transmissions. IRD1 is powered from a 5V supply that is filtered using a 100Ω series resistor and 100µF and 100nF capacitors.

IC1 has two 100nF supply bypass capacitors, for its 3.3V V_{DD} and AV_{DD} lines, plus a 10µF capacitor on the output of its internal 2.5V regulator at pin 20 (V_{CAP}). A 10µF SMD monolithic ceramic capacitor is the preferred type here, as it has good performance and a long life. A 10µF through-hole tantalum capacitor could also be used, and this is catered for in the PCB design.

Power indicator/acknowledge LED1 is switched from output RA0 (pin 2) of IC1. It's fed via a 220Ω series current-limiting resistor, giving an LED current of around 5mA when it is on, ie, when RA0 is driven low.

Power supply

The incoming 12-24V DC supply is applied to either DC socket CON9 (up to 3A) or to pluggable terminal block CON10 (up to 10A). The positive line is fed to the 16 LED strips via a 10A fuse, while a 220µF low-ESR electrolytic capacitor prevents the supply voltage from drooping too much when the LED strips are switched on in unison.

Current also flows from CON9/CON10 to 5V regulator REG1 via reverse polarity protection diode D1. The 5V output from REG1 powers infrared receiver IRD1, buffer op amp IC4, serial latches IC5 and IC6 and the analogue section of DAC IC2.

REG2, a 3.3V low-dropout regulator, is also fed from 5V and supplies power to the remaining components: microcontroller IC1, clock divider IC3, the SD card (via transistor Q10), LED1 and the digital section of DAC IC2.

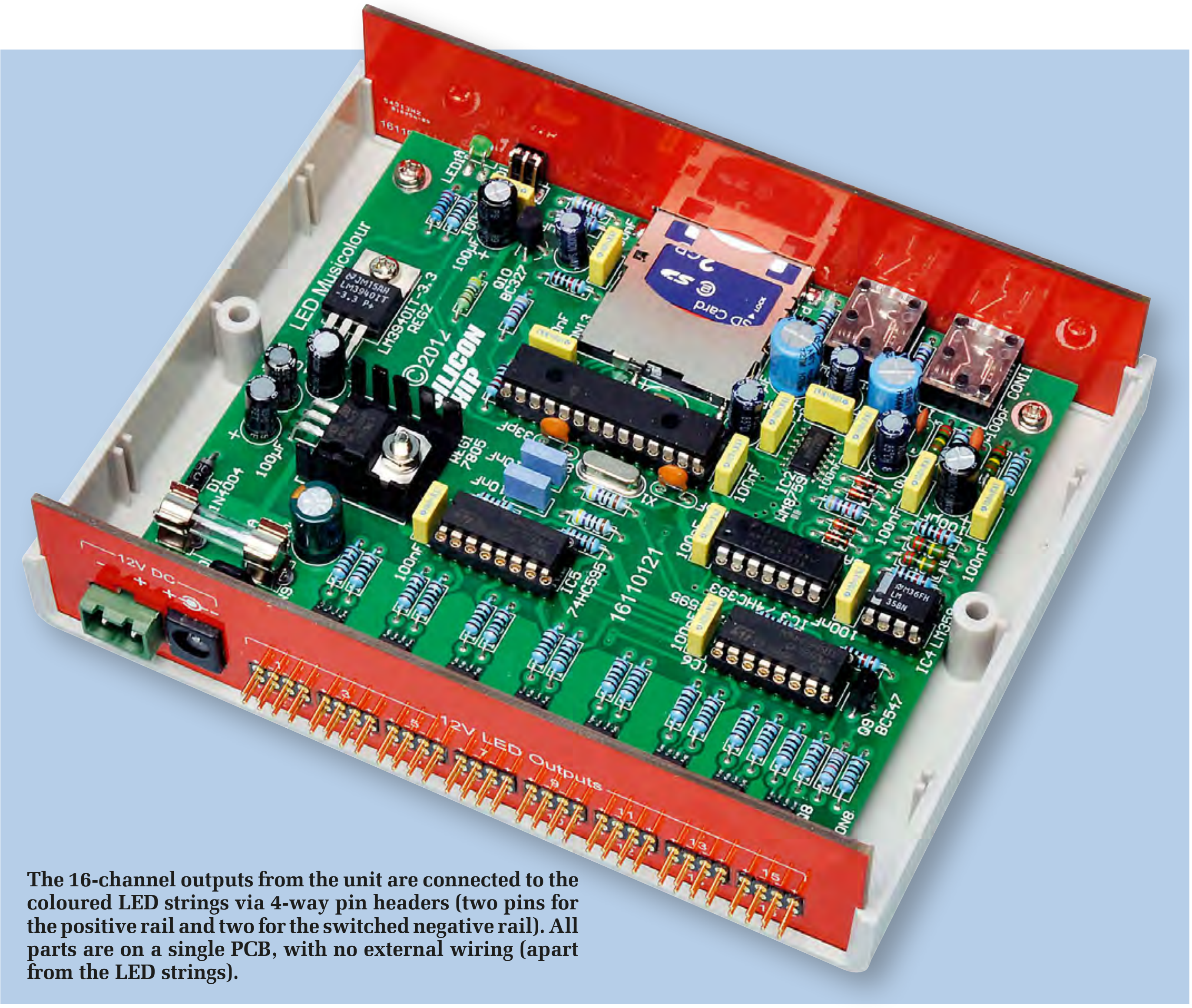
It also acts as the half-supply bias generator for the analogue inputs (IC4a and IC4b).

Software

For those interested, the source code will be available for download from the *EPE* website. We won't go into detail here, but will just give some basic information on its operation.

IC1 only needs to shift one word of data (ie, 16 bits) to IC5/IC6 in order to update the state of all LED outputs. IC5 and IC6 are cascaded, with IC5's Q'H output going to IC6's serial input. This is triggered off one of IC1's internal timers that is set to a rate of 200Hz.

When the timer interrupt occurs, the SPI1 peripheral is used to shift a new word to IC5 and IC6, turning on any LEDs that have a brightness above zero. A second timer is then set to generate an interrupt at the time when the dimmest LED strip needs to be turned off for the correct duty cycle; this delay is calculated as 5ms × (duty cycle) ÷ 100.



The 16-channel outputs from the unit are connected to the coloured LED strings via 4-way pin headers (two pins for the positive rail and two for the switched negative rail). All parts are on a single PCB, with no external wiring (apart from the LED strings).

When this second interrupt is triggered, the handler routine turns off that LED strip and any others with an identical brightness, then re-schedules the timer for the next dimmest LED strip and so on. As a result, depending on the exact brightness value for each LED output, up to 200 × 16 = 3200 interrupts per second are needed to control the LEDs. Thus the overhead is relatively low, given that IC1 runs at around 35MHz.

The LED brightness is updated for every 1024 audio samples. A 1k Fast Fourier Transform (FFT) is then applied, with a Blackman-Harris window, to convert the time domain data to the frequency domain. The magnitudes of the resulting vectors are calculated, giving the frequency content for each bins and the bins are averaged in bands to give the brightness values for the LED outputs.

Audio playback

Since IC2 (the stereo DAC) has no volume control, we need to be able to digitally attenuate the audio data sent to it. Thus, we send 24-bit audio data, even though the WAV files only store 16-bit samples. They are converted to 24 bits by multiplying them by the 8-bit volume level.

For 48kHz audio, IC1's internal clock is set to 36.857MHz using its PLL (8MHz × 129 ÷ 28). MCLK is set to this frequency divided by four, ie, 9.214MHz and BCLK = 9.214MHz ÷ 4 = 2.304MHz. This is the rate at which audio data is serially transmitted to DAC IC2.

The DCI (data converter interface) is set to I^{2}S mode with 24 bits per sample, giving us a sample clock (LRCK) of 2.304MHz ÷ 24 ÷ 2 = 47.991kHz, which is very close to the target of 48kHz.

Getting the DCI to transmit 24-bit data is a little tricky because it works with 16-bit words. We set it to transmit two 12-bit words per sample and the audio data is stored in memory in 32-bit chunks, with eight bits of each unused.

The rest of the software is straightforward and re-uses much of our existing dsPIC33 codebase, including the SD card interface, FAT16/32 file system layer, interrupt-based infrared protocol decoding and so on. Besides the LED strip control, the main area of new code for this project is the audio playback layer, which has been enhanced to support WAV files residing in multiple folders. It also supports folder hierarchies several levels deep.

That's all we have space for this month. Next month, we will describe the assembly and show you how to drive it.

Stranger than fiction

TechnoTalk

Mark Nelson

Even if Superman is a fictional character, two marvels from the films are now entirely real, with one having remarkable and longlasting potential. Plus, who's driving your car when you're behind the wheel? That's not such a daft question, as Mark Nelson reports

BACK in 2007, a new mineral was discovered in Serbia that had the same chemical formula as the one written on a case of 'kryptonite' stolen from a museum by super-villain Lex Luthor in the film *Superman Returns*. Sadly for Sci-Fi fans, the constituent elements are all that the real mineral shares with its comic book counterpart. If that's a disappointment, something else that Lex Luthor stole in the same film – 'memory crystal' – is real and could have extremely valuable applications. I haven't seen the film, so I don't know how these crystals would have assisted Luthor in his nefarious plans, but in their new and real incarnation the crystal's potential is enormous.

'Crystal memory could outlast civilisation'

Who could resist reading a story that begins like that? Especially when it continues: 'A computer memory crystal that has been developed in the UK could end up outliving the entire human race and survive for one million years without any information being lost', according to some experts.

It may sound fantastic, but that's what was proudly announced in July by Southampton University. Scientists there have devised a process for laser writing five-dimensional digital data onto nanostructured glass. Dubbed the 'Superman memory crystal', the glass memory has been compared directly to the 'memory crystals' in Superman tales. Data is recorded via self-assembled nanostructures created in fused quartz, which is able to store vast quantities of data for potentially more than a million years. The information encoding is realised in five dimensions: the size and orientation in addition to the three-dimensional position of these nanostructures.

The research team has, for the first time anywhere, experimentally demonstrated the recording and retrieval processes. This method of data storage offers unprecedented capabilities, including 360-terabyte capacity per disc, thermal stability up to 1,000°C and practically unlimited lifetime. A 300Kb text file has been recorded successfully using an ultrafast laser, producing extremely short and intense pulses of light. The file is written in three layers of nanostructured dots separated dimensionally by five micrometres (five millionths of a metre). The pattern produced changes the way light travels through glass, modifying the polarisation of light that can then be read by a combination of an optical microscope and a polariser similar to that found in sunglasses.

Long-term stability

Practical electronicists like us are unlikely to need storage of this scale, but it could be ideal for museums and major archives that have huge numbers of documents and photographs that they need to preserve. Project leader Jingyu Zhang from the university's Optoelectronics Research Centre (ORC) explained. 'We are developing a very stable and safe form of portable memory using glass, which could be highly useful for organisations with big archives. At present, companies have to back up their archives every five to ten years because hard-drive memory has a relatively short lifespan,' he said. Leader of the research centre, prof Peter Kazansky, added: 'It is thrilling to think that we have created the first document which will likely survive the human race. This technology can secure the last evidence of civilisation: all we've learnt will not be forgotten.'

Commercialisation of this ground-breaking new technology is the next step, and the team is looking for industry partners. The work is being conducted jointly with Eindhoven University of Technology in the Netherlands, within the framework of the EU's Femtoprint initiative.

The Web in your wheels

'My Mercedes has an Internet connection I was completely unaware of until I became suspicious that the clock was so bang on time, and then I noticed when I put a CD in that it checked the Gracenotes database to get the track names.'

This recent thread on a chat-list caught my eye for two reasons. First, I was unaware that cars could be 'wired' in this way (it makes sense, but hadn't occurred to me) and second, I was surprised that someone could drive a car with this feature without having requested it – perhaps the writer's employer supplied the vehicle. Looking at the Mercedes-Benz website, I see that an 'InCar Hotspot' is provided on several of the company's upmarket models and provides 'everything you need for unrestricted Internet access in the vehicle' at data speeds of up to 7.2Mbit/s. You can connect laptops, netbooks, smartphones or tablet PCs such as the iPad, while the special WLAN router in the vehicle's boot allows passengers to surf the Internet during the journey using up to four devices at the same time.

This is all very nice if you can afford it – or is it? Conspiracy theorists are having a field day over the death earlier this year of *Rolling Stone* magazine contributing editor Michael Hastings, whose car was a 2013-model Mercedes-Benz C250. Richard Clarke, a former chief counter-terrorism adviser on the US National Security Council, has stated that Hastings' death was 'consistent' with a cyber-attack on his car. He said that not only does the technology to hack cars exist, but 'there is reason to believe that intelligence agencies for major powers, like the United States, are already equipped to stage such an attack.'

Easy to hack

He told the *Huffington Post*: 'What has been revealed as a result of some research at universities is that it's relatively easy to hack your way into the control system of a car, and to do such things as cause acceleration when the driver doesn't want acceleration, to throw on the brakes when the driver doesn't want the brakes on, or to launch an air bag. You can do some highly destructive things now, through hacking a car, and it's not that hard. So if there were a cyber-attack on the car – and I'm not saying there was, I think whoever did it would probably get away with it.'

The reason why it's 'not that hard' is because many modern cars employ an engine control unit (ECU) that controls engine power, transmission and braking. Spanish 'hardware hackers' Javier Vázquez Vidal and Alberto Garcia Illero have already demonstrated a device costing £16 to make that can bypass the security encryption of ECUs. This currently needs to be installed on the target car, but the next version will be wireless. The report on the **Planet.Infowars.com** website sums up the situation neatly, 'Whatever the causes of Michael Hastings's crash, the need to make cars secure against hacking will only become more acute.'

Win a PIC32 GUI Development Board with Projected Capacitive Touch

EVERYDAY PRACTICAL ELECTRONICS is offering its readers the chance to win a PIC32 GUI Development Board with Projected Capacitive Touch (#DM320015). This PIC32 GUI Development Board enables development of cost-effective multi-touch graphical user interfaces. It is based on PIC32MX795F512H with 105 DMIPS performance, 512Kb Flash and 128Kb RAM. The PIC32 is coupled with low-cost PSRAM for high speed graphics frame buffering and a 4.3-inch WQVGA touch display enabling development of graphics solutions without an external graphics controller.

Additionally, the board provides USB host and device connectivity and supports I/O connections via through-hole pads for custom board attachment. Multi-touch user input is supported by Microchip's Turnkey Projected Capacitive Touch Controller, MTCH6301. The board is a standalone development platform that can be programmed/debugged via the on board 5-pin in-circuit serial programmer interface designed for Microchip's PICkit3 in-circuit debugger.

Key features
Based on a PIC32MX795F512 device with 512Kb Flash and 128Kb RAM, Projected Capacitive Touch device MTCH6301, WQVGA 4.3-inch display, USB port for device or host functionality, expansion header.

HOW TO ENTER

For your chance to win a PIC32 GUI Development Board with Projected Capacitive Touch, please visit: **www.microchip-comps.com/epe-gui** and enter your details in the entry form.

CLOSING DATE

The closing date for this offer is 31 October 2013

Measure and control temperature over a wide range with this . . .

By JOHN CLARKE

High-Temperature Thermometer/Thermostat

Need to measure or control temperature over a very wide range? Now you can do it with this compact unit which hooks up to a K-type thermocouple. It drives a relay which can be used to precisely control the temperature in ovens, kilns, autoclaves, solder baths or at the cold end of the 'spectrum', fridges and freezers. It is based on an Analog Devices AD8495 precision instrumentation amplifier with thermocouple cold junction compensation.

SOME DIGITAL MULTIMETERS can measure temperature with a K-type thermocouple, but that's all they can do; they cannot control the temperature in an oven or kiln. In other words, they do not provide an adjustable thermostat function. However, our new *High-Temperature Thermometer/Thermostat* can be used to measure and control the temperature at the same time. That's because it has a relay output that opens or closes at a preset temperature.

The switched output can be used directly or in conjunction with a higher-rated relay to control power to the element of a heater or the compressor of a refrigerator. For heating, the power can be switched on when the temperature is below the preset temperature and switched off when it is above. Alternatively, for cooling, power can be switched on when the temperature goes above the preset and off when it goes below. The preset temperature for this thermostat action can be adjusted between –50°C and 1200°C.

It is important that the thermostat function does not cause rapid on and off switching of the heater, compressor or whatever is being used to control the amount of heating or cooling. Hence the design incorporates adjustable hysteresis. This allows a preset temperature difference to apply between switching power on and off. The hysteresis is adjustable from less than 1°C to more than 9°C.

The temperature is displayed on a 3½ digit LCD, and while the unit can display a temperature from –50°C to 1200°C, the actual measurement range will depend on the particular probe. Some K-type probes will operate from

Features and specifications

Main features

- K-type thermocouple probe
- Ground-referenced or insulated probe can be used
- Measures –50°C to 1200°C (depending on probe)
- Pre-calibrated temperature measurement
- Optional calibration of span and offset adjustment
- Thermostat switching at a preset temperature with adjustable hysteresis
- High-to-low or low-to-high thermostat threshold
- Relay output for thermostatic control
- Relay contacts rated at 10A (30V AC/DC maximum recommended switching voltage)

Specifications

Power supply: 12V @ 100mA

Measurement range: –50°C to 1200°C (probe dependent)

Initial accuracy: ±4°C for –25°C to 400°C measurements (ambient between 0°C and 50°C)

Optional calibration adjustment for span: –4%, +5.27%

Optional calibration adjustment for offset: ±6.2mV equivalent to >±1°C

Thermostat set point range: adjustable from –50°C to 1200°C

Thermostat hysteresis: adjustable from <1°C to >9°C

Cold-junction compensation: optimised for 0-50°C ambient temperatures

–50°C to 250°C, while others operate from –40°C to 1200°C.

The *High-Temperature Thermometer/Thermostat* is housed in a small instrument case and its controls on the front include a power switch and a switch to select between measured temperature and the preset thermostat temperature. An LED indicator is for power indication and a second LED shows when the thermostat relay has switched on.

At the rear of the case is the power input socket for a 12V DC supply and a socket for the K-type thermocouple. Additionally, there is a terminal connector inside the case for connection to the thermostat relay contacts. The common (C), normally open (NO) and normally closed (NC) contacts are available for connection.

Inside the case there are jumper links to select whether the thermostat relay switches on above or below the preset temperature for the thermostat. There are also jumper selections to select whether the *Thermometer/Thermostat* is built pre-calibrated or where the temperature calibration can be accurately adjusted.

K-type thermocouple

This design uses a K-type thermocouple, which comprises a junction of two dissimilar wires; in this case an alloy of chrome and nickel (called chromel) for one wire and an alloy of aluminium, manganese, silicon and nickel (called alumel) for the second.

These two wires are insulated and make contact at the temperature probe end only. The other end of the wires are usually connected to a 2-pin plug.

Basically, a thermocouple's operation relies on the principle that the junction of two dissimilar metals produces a voltage that is dependent on temperature. A K-type thermocouple produces a voltage output that typically changes by 40.44µV/°C. This change in output is called the Seebeck coefficient, and it refers to the output change that occurs due to the temperature difference between the probe end and the plug end of the thermocouple.

In practice, the Seebeck coefficient for the K-type thermocouple varies with temperature and is not precisely 40.44µV, but this is a good average value over the temperature range from 0°C to 1200°C.

If we know the temperature at the plug end of the thermocouple, we can calculate the temperature at the probe since we also know the Seebeck coefficient. For example, if the plug end is held at 0°C, then the output will increase by 40.44µV for every 1°C increase. Similarly, the output will decrease by 40.44µV for every 1°C drop in temperature.

In practice, we do not keep the plug end of the thermocouple at 0°C; it's not practical. Instead, we compensate the thermocouple output by measuring the temperature at the plug end and then adding 40.44µV for every 1°C that the thermocouple plug end is above 0°C or subtracting 40.44µV for every 1°C that the plug end is below 0°C.

For example, if the thermocouple plug is at 25°C, its output will be 1.011mV (ie, 25 x 40.44µV) lower than it would be if it were at 0°C. By adding an extra 1.011mV to the reading, we obtain the correct result without having to keep the plug end at 0°C.

Note that there are several dissimilar metal junctions within the connections between the thermocouple plug and amplifier. These include the Chromel to copper junction and the Alumel to copper junction on the PCB itself. These do not contribute to the overall voltage reading *provided they are all kept at the same temperature.*

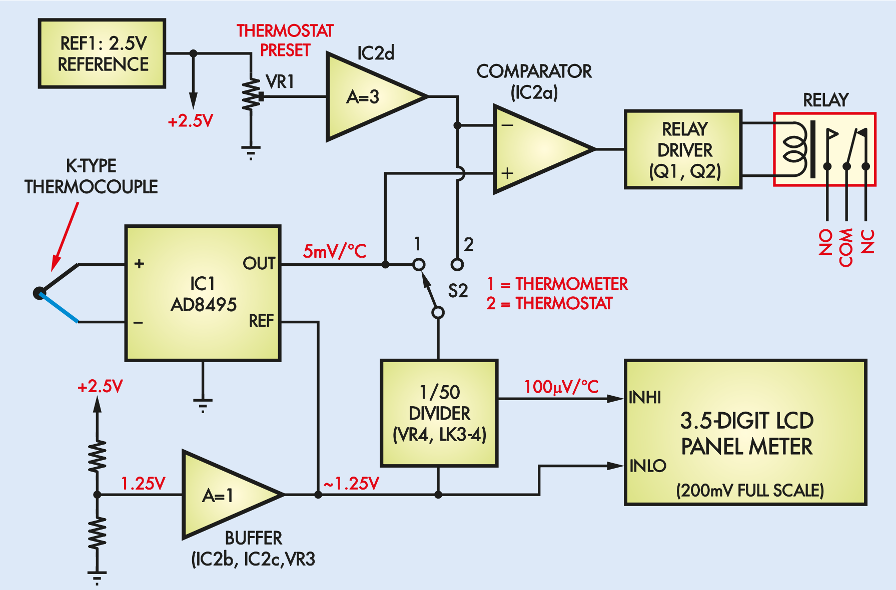


Fig.1: block diagram of the *High-Temperature Thermometer/Thermostat.* IC1 processes and amplifies the thermocouple's output and drives the LCD panel meter and comparator IC2a. Trimpot VR1 sets the thermostat temperature

Therefore, the PCB has been designed to help maintain similar temperatures at these junctions by making the copper connections all the same size. Once the PCB is installed inside its case, the inside temperature should remain fairly constant for all these junctions.

Note that if the thermocouple lead needs to be extended, it's necessary to use the same K-type thermocouple wire for the whole length between the probe and plug – including the materials used for any additional plugs and sockets.

Signal processing

Refer now to Fig.1, which shows the block diagram of the *High-Temperature Thermometer/Thermostat*. As shown, the thermocouple signal is processed using the Analog Devices AD8495 IC. This is a precision instrumentation amplifier with K-type thermocouple cold junction compensation. Its output is 5mV/°C.

The amplifier within the AD8495 is laser trimmed for a gain of 122.4. This gain effectively converts the 40.44µV/°C output of the thermocouple to 4.95mV/°C. The output is optimised for a 25°C measurement, where a gain of 122.4 gives a result of 123.75mV.

Within the AD8495, a 1.25mV offset is added to the amplified value, giving a 125mV output at 25°C. For temperatures other than 25°C, the combination of the variation in the Seebeck coefficient over temperature, the 122.4 gain and the 1.25mV offset provides an accurate 5mV/°C output over the range of –25°C to 400°C. For this range, the output is within 2°C.

Note that the specification panel shows that the accuracy is ±4°C for ambient between 0°C and 50°C and –25°C to 400°C measurements. This is different to the 2°C error for the AD8495 because the display is showing a reading via a voltage divider that is prone to extra tolerance errors.

It's possible to calibrate the measurement to a finer accuracy if this is required. Table 1 shows the expected output from the AD8495 over a wide range of temperatures and compares this with the ideal 5mV/°C output.

How it works

Returning now to the block diagram of Fig.1, the K-type thermocouple connects directly to the AD8495 (IC1) at the IN+ and IN– terminals. The resulting 5mV/°C output signal from IC1 is then fed to the non-inverting input of comparator IC2a and also to position 1 (Temperature) on switch S2. S2 selects between the temperature and thermostat modes of operation.

In order to allow for negative temperature measurements, the output from the AD8495 is offset by approximately 1.25V. This offset is derived by a voltage divider connected across a 2.5V reference (REF1) and buffered using op amps IC2b and IC2c. The buffered 1.25V signal is then applied to the AD8495's REF (reference) input.

This effectively 'jacks up' the AD8495's output by 1.25V. As a result, a –50°C measurement now gives an output that's theoretically 250mV below (–5mV × 50) the 1.25V reference offset (ie, 1V). Without this offset, the AD8495 would not be able to handle negative temperature measurements since its output cannot go below 0V.

Although the offset only needs to be 250mV to allow for a –50°C measurement, a value of 1.25V is used because of the LCD panel meter that's used to measure the voltage. This meter requires an input that's at least 1V above the 0V supply for correct operation. According to Table 1, the actual output from IC1 at –50°C is 228mV below the offset voltage. So using an offset of 1.25V leaves us with a comfortable 22mV margin above the critical 1V level.

The 3½-digit LCD panel meter used to display the temperature has a 200mV full-scale reading (actually 199.9mV) for a reading of 1999. It's basically connected to measure the voltage between IC1's output (via a divider) and the offset voltage. This effectively removes the offset voltage from the reading.

To prevent the meter from overranging and to get a reading in °C, we need to divide IC1's output by 50. For example, if the temperature is 1200°C, the voltage between IC1's output and the 1.25V offset will be 6V (ie, 1200 × 5mV). Dividing this by 50 gives 120.0mV and the panel meter is configured to show 1200 (no decimal point).

Note that, in the full circuit, either a fixed divide-by-50 attenuator or an adjustable divide-by-50 attenuator can be used. The desired attenuator is selected using jumper links and the adjustable one allows for accurate calibration.

The display can either show the measured temperature when switch S2 is in position 1 or the preset temperature (for the thermostat operation) when S2 is in position 2. VR1 sets the thermostat temperature. As shown, it's connected to a 2.5V reference (REF1) and the voltage at its wiper drives op amp IC2d.

As a result, IC2d's output can range up to 7.5V, slightly more than the 7.25V at IC1's output when the measured temperature is at the 1200°C maximum (ie, 1200 × 5mV plus the 1.25V offset). This allows VR1 to set the thermostat temperature anywhere from –50°C to 1200°C.

IC2d's output is fed to the inverting input of comparator IC2a, where it is

compared with IC1's output. IC2a's output thus switches low when the temperature is below the preset and high when the temperature is above the preset. This output then drives a relay via transistors Q1 and/or Q2.

Links LK5 and LK6 can be selected so that the relay either switches on when IC2b's output goes high or on when it goes low.

Circuit details

Refer now to Fig.2 for the full circuit diagram of the *High-Temperature Thermometer/Thermostat.* As well as the AD8495 (IC1) and the LCD panel meter, it includes an OP747 precision quad op amp (IC2), a 7805 3-terminal regulator, an LM285-2.5 precision voltage reference, transistors Q1 and Q2 and various minor components.

IC1 is powered from a 12V DC plugpack supply via switch S1, diode D1 (for reverse polarity protection) and a 10Ω resistor. A 22V Zener diode (ZD1) clamps any over-voltage transients, while 100µF and 100nF capacitors are used to bypass the supply.

In operation, IC1 draws just 180µA to minimise internal heating (note: internal heating would affect the measurement of the ambient temperature used for the thermocouple 'ice-point' temperature compensation). The K-type thermocouple connects to its IN+ and IN– terminals (pins 8 and 1) via series 47kΩ resistors. These resistors and their associated 100nF ceramic capacitors prevent RF (radio frequency) signals from being detected by IC1's sensitive input stages. The resistors acts as RF stoppers, while the 100nF capacitors effectively shunt any remaining RF signal to ground.

In addition, the negative terminal of the K-type thermocouple is tied to ground via a 100Ω resistor. This prevents the probe from picking up noise and mains hum, which would cause erratic operation.

Note that the 100Ω resistor is included so that the circuit can be used with both earthed and insulated-sheath thermocouples. Basically, the thermocouple probe wires are housed in a cylindrical metal sheath or rod. Some units connect the negative thermocouple wire directly to this metal sheath (an earthed probe), while others fully insulate the metal sheath from the thermocouple wires (an insulated probe).

For an insulated probe, it doesn't matter whether the negative terminal

Table 1: AD8495 output versus temperature

Thermocouple Temperature (°C)	Ideal Output @ 5mV/°C (mV)	AD8495 Output (mV)	Display Reading (°C) ±1 Digit
-50	-0.25	-0.228	-46
-40	-0.2	-0.184	-37
-20	-0.1	-0.093	-19
0	0.0	0.003	0
20	0.1	0.100	20
25	0.125	0.125	25
40	0.2	0.200	40
60	0.3	0.301	60
80	0.4	0.402	80
100	0.5	0.504	101
120	0.6	0.605	121
140	0.7	0.705	141
160	0.8	0.803	161
180	0.9	0.901	180
200	1.0	0.999	199
220	1.1	1.097	219
240	1.2	1.196	239
260	1.3	1.295	259
280	1.4	1.396	279
300	1.5	1.497	299
320	1.6	1.599	320
340	1.7	1.701	340
360	1.8	1.803	361
380	1.9	1.906	381
400	2.0	2.010	402
420	2.1	2.113	423
440	2.2	2.217	443
460	2.3	2.321	464
480	2.4	2.425	485
500	2.5	2.529	506
520	2.6	2.634	527
540	2.7	2.738	548
560	2.8	2.843	569
580	2.9	2.947	589
600	3.0	3.051	610
620	3.1	3.155	631
640	3.2	3.259	651
660	3.3	3.362	672
680	3.4	3.465	693
700	3.5	3.568	713
720	3.6	3.670	734
740	3.7	3.772	754
760	3.8	3.874	774
780	3.9	3.975	795
800	4.0	4.076	815
820	4.1	4.176	835
840	4.2	4.275	855
860	4.3	4.374	875
880	4.4	4.473	895
900	4.5	4.571	914
920	4.6	4.669	934
940	4.7	4.766	953
960	4.8	4.863	973
980	4.9	4.959	992
1000	5.0	5.055	1011
1020	5.1	5.150	1030
1040	5.2	5.245	1049
1060	5.3	5.339	1068
1080	5.4	5.432	1086
1100	5.5	5.525	1105
1120	5.6	5.617	1123
1140	5.7	5.709	1141
1160	5.8	5.800	1160
1180	5.9	5.891	1178
1200	6.0	5.980	1196

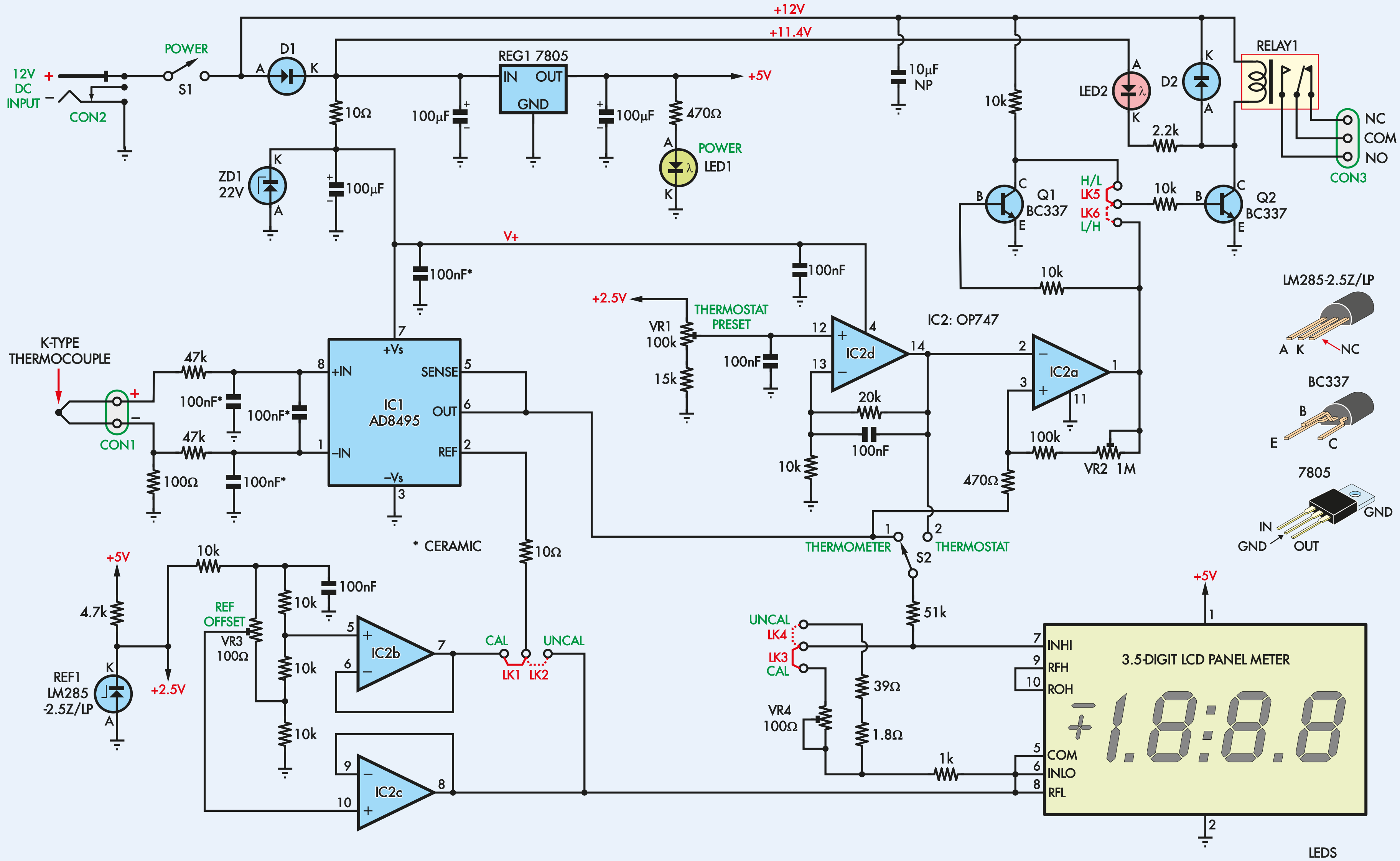


Fig.2: the complete circuit diagram for the *High-Temperature Thermometer/Thermostat*. An accurate 1.25V reference is derived from REF1 via IC2b or IC2c and this is applied to the REF input of IC1 to enable measurements down to –50°C. IC1's output drives the LCD panel meter via a 50:1 divider and also drives the non-inverting input of comparator IC2a. IC2a compares IC1's output with the thermostat preset temperature, as set by VR1 and IC2d, and drives relay 1 via transistors Q1 and/or Q2 when the preset limit is reached. Links LK5 and LK6 allow the relay to be driven on either a rising or falling temperature

is connected directly to ground or connected to ground via a 100Ω resistor. That's because an insulated probe can connect to a point that's not at 0V without affecting the operation of the probe.

By contrast, an earthed probe does require the 100Ω resistor. That's because the probe could make an external connection to the 0V supply rail and this might not be at exactly the same voltage as the 0V rail inside the unit.

This type of situation could easily arise, for example, when measuring engine heat or brake disc heat in a car and the unit is being powered by the vehicle's battery. In this situation, the probe point and the internal 0V rail will be at slightly different voltages due to current flowing in the vehicle's chassis. The difference in voltage may only be small but the thermocouple's output only varies by about 40µV/°C, so only small variations can mean a huge error in temperature readings.

The 100Ω resistor eliminates this problem by preventing significant current flow between the thermocouple's negative terminal and the 0V rail within the thermometer.

Deriving the offset

The 1.25V offset for IC1 is derived from REF1, a precision 2.5V voltage reference, via a resistive divider. This divider comprises four 10kΩ resistors and a 100Ω trimpot (VR3).

As shown, the 1.25V midpoint of the 10kΩ fixed resistive divider is fed to pin 5 of IC2b, while the voltage on VR3's wiper is fed to IC2c. VR3 allows the offset voltage to be varied over a small range either side of 1.25V.

IC2b and IC2c are both connected as unity-gain buffer stages. When LK1 is installed, IC2b provides a fixed 1.25V offset for IC1 at its REF (pin 2) input. At the same time, IC2c provides the variable offset output to the panel meter at its IN LO input.

Alternatively, if LK2 is installed, IC2c drives both the reference input of IC1 and the INLO input of the LCD panel meter. In this case, the voltage applied to both IC1's REF input and the panel meter's INLO input are exactly the same and this is the linking option to use if you do not want to accurately adjust the temperature calibration.

LCD panel meter

As stated previously, the LCD panel meter measures the difference between its INHI (pin 7) and INLO (pin 6) inputs. In this circuit, IC1 drives the INHI input via one of two 50:1 voltage dividers (one fixed, the other variable) when S2 is in position 1. IC1 is capable of delivering in excess of ±5mA to a load, but the fixed 50:1 divider draws just 115µA maximum when IC1's output is producing 7.25V for a 1200°C measurement. This low current minimises any internal heating of the IC.

The fixed divider is selected using link LK4. It's made up using a 51kΩ resistor in the top section and 39Ω, 1.8Ω and 1kΩ resistors at the bottom. Assuming the values are exact, the division ratio is very close to 50:1. However, resistor tolerances can shift this to within a range of around 50.05:1 to 49.95:1.

The variable divider shares the 51kΩ and 1kΩ resistors but uses a 100Ω trimpot in place of the 39Ω and 1.8Ω resistors in the fixed divider. This allows the divider to be adjusted. It's selected by installing link LK3 instead of LK4.

The LCD panel meter itself is based on an Intersil ICL7106 3½-digit LCD analogue-to-digital converter (ADC). Its INLO, COM (common) and RFL (reference low) pins are all connected together; ie, they are all fed with the reference offset voltage at IC2c's output. In addition, the ROH output is connected to the RFH (reference high) input, and this sets the panel meter to 200mV full scale.

A 5V supply rail for the LCD is derived from regulator REG1 (7805). REG1's input and output rails are both filtered using 100µF electrolytic capacitors, while LED1 in series with a 470Ω current-limiting resistor provides power indication.

This regulated 5V supply also drives the 2.5V reference (REF1), this time via a 4.7kΩ resistor. As well as providing a source for the offset voltage, the resulting 2.5V rail is also fed to the top of VR1 which sets the thermostat preset.

VR1 is connected in series with a 15kΩ resistor across this supply and its wiper provides an output which ranges from 326mV up to 2.5V. IC2d amplifies this by three, as set by the 20kΩ and 10kΩ resistors in the feedback path. The resulting voltage at the output of IC2d can range anywhere from 978mV up to 7.5V and that more than covers the possible voltage range from IC1, for temperatures ranging from –50°C to 1200°C.

As described previously, op amp IC2a is wired as a comparator. It monitors IC2d's output and compares this with IC1's output. IC2a thus switches its output high when the measured temperature is above the preset temperature, or low when the measured temperature goes below the preset (ignoring hysteresis).

Trimpot VR2 (1MΩ) and the 100kΩ and 470Ω resistors provide hysteresis. With VR2 set at 1MΩ, the hysteresis is at its minimum and there is less than 1°C hysteresis. At the other extreme, with VR2 set for 0Ω, the hysteresis is more than 9°C.

Relay driver circuit

IC2a drives transistor Q1, which in turn drives Q2, when link LK5 is inserted. Alternatively, if LK6 is selected, Q1 is bypassed and IC2a drives Q2 direct.

These two links select whether the relay turns on for a low-to-high temperature transition (LK6 in place) or a high-to-low transition (LK5 in place). When LK6 is in circuit, Q2 turns on when IC2a's output goes high (ie, when the temperature rises above the preset) and this turns on relay 1. The relay subsequently turns off again when IC2a's output switches low (ie, when the temperature falls below the preset).

Conversely, when LK5 is in circuit, Q1 inverts the logic. In this case, Q2 and the relay are normally on since Q2's base is pulled high. However, when IC2a's output switches high (as the temperature rises above the preset), Q1 turns on and pulls Q2's base to ground. Thus, Q2 and the relay turn off and remain off until the temperature falls below the preset again.

LED2 lights whenever the relay switches on to indicate that the set

The OP747ARZ quad precision op amp

The OP747ARZ quad precision op amp specified here has features that are not found in general-purpose op amps.

First, it features a low offset voltage of 100µV maximum and the input bias and offset currents are in the very low nA range. Second, it can handle input voltages ranging from the ground supply rail up to within 1V of the positive supply. And third, the output can reach close to each supply rail.

Taken together, these characteristics make this quad op amp ideal for this circuit.

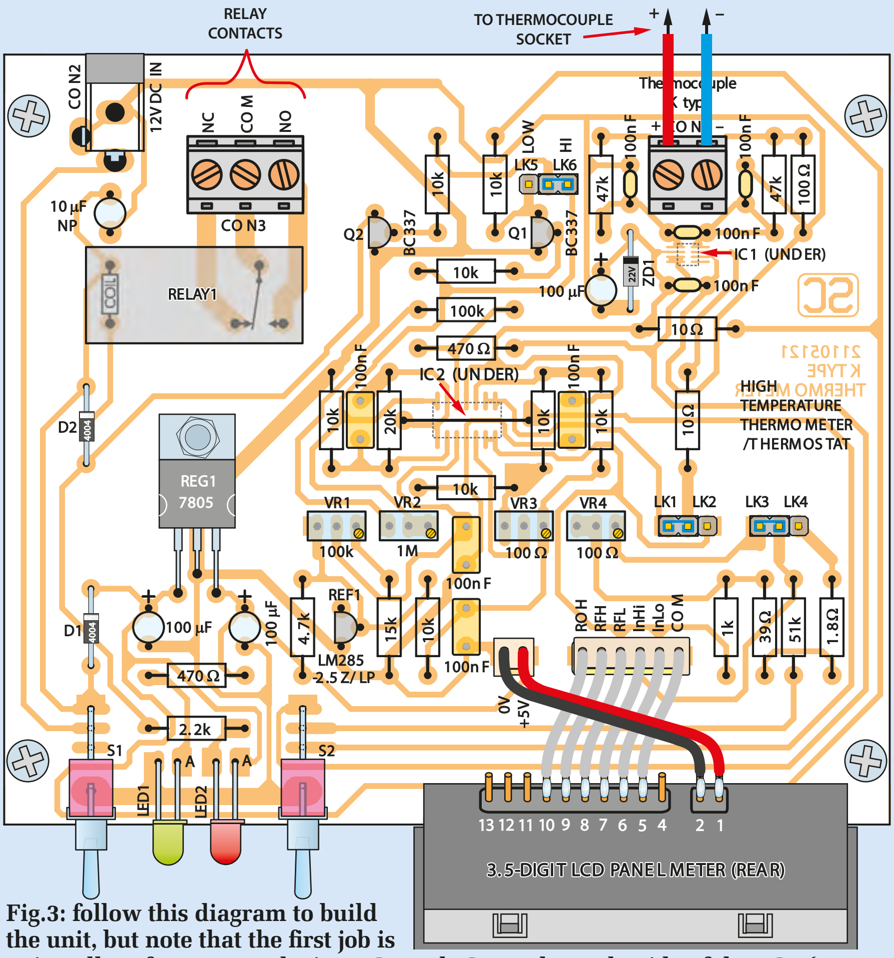


Fig.3: follow this diagram to build the unit, but note that the first job is to install surface-mount devices IC1 and IC2 on the underside of the PCB (see below). You can omit the relay, CON3, S2 and transistors Q1 and Q2 if you intend using the unit as a thermometer only and don't need the thermostat function

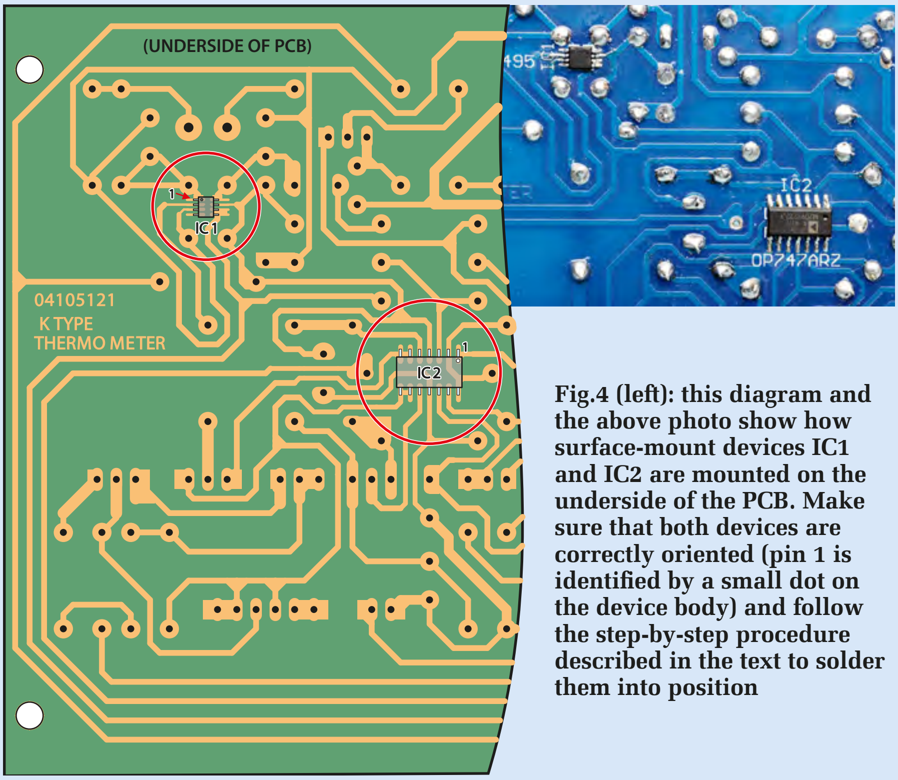


Fig.4 (left): this diagram and the above photo show how surface-mount devices IC1 and IC2 are mounted on the underside of the PCB. Make sure that both devices are correctly oriented (pin 1 is identified by a small dot on the device body) and follow the step-by-step procedure described in the text to solder them into position

temperature threshold has been reached. The associated 2.2kΩ resistor limits the current through LED2, while diode D2 protects Q2 from damage by quenching the back-EMF coil voltage spikes generated when the relay turns off.

The relay provides the usual common (COM), normally open (NO) and normally closed (NC) contacts, so it can drive a load on or off depending on the selection of the NO or NC contacts.

So it may seem that links LK5 and LK6 are not really necessary to reverse the switching sense. However, there are reasons why you may wish to select whether the relay is normally powered or not, especially when the relay contacts are required to switch a heating or cooling operation.

One reason is that less current is drawn by the circuit when the relay is off and you might want to choose the link and contact configuration that draws the least power.

Another reason is that you might want to ensure fail-safe operation if power is cut to the circuit. By using the COM and NO contacts to do the switching, you can ensure that power is not provided for heating or cooling if the power to the *Thermometer/Thermostat* fails.

Construction

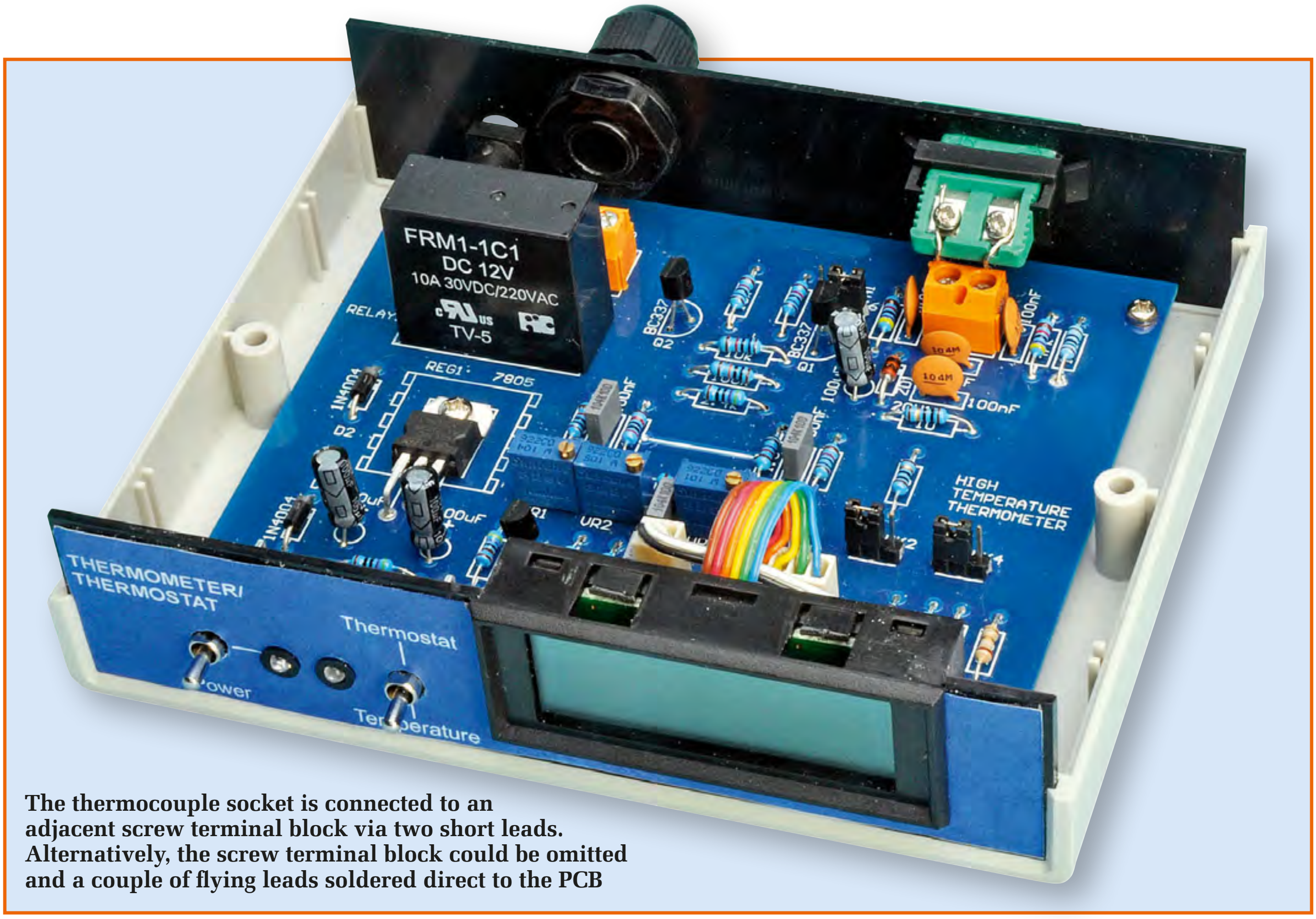
The assembly is straightforward, with all parts except the probe socket and the LCD panel meter mounted on a PCB coded 21105121 (117mm × 102mm). This is housed in a plastic instrument case measuring 140mm × 110mm × 35mm.

Begin by carefully checking the PCB for any defects. Check also that the hole sizes are correct for each component to fit neatly. The corner mounting holes and the regulator mounting hole should all be 3mm in diameter.

Our prototype used a double-sided PCB and Fig.3 shows the parts layout. The first step is to install IC1 and IC2. These are both surface-mount devices (SMDs), which mount on the underside of the PCB – see Fig.4.

To install these, you will need a fine-tipped soldering iron, some fine solder and some quality solder wick. A magnifying lamp or at least a magnifying lens will also be handy.

It's best to install IC2 first. This is the 14-pin device with the wider pin spacings. First, place the PCB copper side up and apply a small amount of solder to the top-right pad, then pick the IC up with tweezers and position



The thermocouple socket is connected to an adjacent screw terminal block via two short leads. Alternatively, the screw terminal block could be omitted and a couple of flying leads soldered direct to the PCB

it near the pads. Check that it is oriented correctly (ie, with its pin 1 dot positioned as shown on Fig.4), then heat the tinned pad, slide the IC into place and remove the heat.

Now check the IC's alignment carefully using a magnifying glass. It should be straight, with all the pins centred on their respective pads and an equal amount of exposed pad on either side. If not, reheat the soldered pin and nudge the chip in the right direction.

Once its position is correct, solder the diagonally opposite pin, then recheck its position before soldering the remaining pins. Don't worry too much about solder bridges between pins at this stage; they are virtually inevitable and can easily be fixed. The most important job right now is to ensure that solder flows onto all the pins and pads.

Once you've finished, apply a thin smear of no-clean flux paste along all the solder joints and remove the excess solder using solder wick. You should then make a final inspection to ensure that there are no remaining solder bridges and that the solder has not 'balled out' onto a pin without flowing onto the pad.

If there are still bridges, clean them up with more flux and solder wick.

Once IC2 is in place, you can install IC1 in exactly the same manner.

Through-hole parts

The larger through-hole parts can now be installed on the top of the PCB. Start with the resistors and diodes, then install Zener diode ZD1, the MKT and ceramic capacitors and the electrolytics. It's a good idea to check the value of each resistor using a multimeter before installing it.

Take care with the polarity of the electrolytics, the diodes and the Zener diode. They must be oriented as shown on Fig.3.

Transistors Q1 and Q2 and the LM385-2.5 precision voltage reference (REF1) can go in next. REG1 can then be installed. This mounts horizontally with its tab against the PCB, so you will have to bend its leads down at right angles to match its mounting holes.

Secure its tab to the PCB using an M3 × 6mm screw and nut before soldering its leads. Don't solder the leads before securing the tab; you could crack the copper tracks as the mounting screw is tightened if you do.

Trimpots VR1-VR4 are next on the list. These must all be mounted with the adjustment screw to the right. Follow with the three 3-way pin headers for links LK1-LK6, then install the 6-way and 2-way polarised headers for the LCD panel meter connections. Be sure to orient these headers as shown, ie, with their vertical tabs towards the panel meter.

Once they're in, you can install the two LEDs, but first you have to bend their leads down through 90° some 9mm from their bodies.

The best way to do this is to first cut a cardboard spacer 9mm wide. This is then be used as a template when bending the LED leads. Make sure that each LED is correctly oriented before bending its leads – the (longer) anode lead must be on the right when looking at the lens.

Having bent their leads through 90°, the two LEDs must be installed with their leads 5mm above the PCB. This is best done by pushing them down onto a 5mm spacer, then soldering the leads to the PCB pads.

The cable gland on the rear panel allows an external lead to be fed into the case and connected to the relay contacts at CON3. The LCD is secured to the front panel by running a couple of beads of silicone adhesive or hot-melt glue down the vertical inside edges.

Switches S1 and S2 are right-angle types and are mounted directly on the PCB. Push them down onto the board as far as they will go before soldering their leads. The PCB assembly can then be completed by installing the relay, the DC socket (CON2) and the 2-way and 3-way screw terminals.

Connecting the panel meter

The panel meter is wired to the 6-way header plug and to the 2-way header plug using short lengths of ribbon cable. These leads can be obtained by separating an 8-way ribbon into 6-way and 2-way strips.

Cut these strips to 50mm in length, then strip about 2mm of insulation from the individual wires at one end and crimp them to the header pins. The pins are then inserted into the headers.

The other ends of these leads can then be stripped and soldered to the LCD panel meter pins. Check carefully to ensure that each wire goes to the correct pin on the panel meter and that there are no shorts between them. In fact, it's a good idea to slip a short length of heatshrink over each wire before soldering it and then pushing over the soldered joint to insulate it from its neighbours.

Jumper links LK2 and LK4 should now be installed and either LK5 or LK6. Install LK5 if you want the relay to switch on when the temperature drops below the preset. Alternatively, install LK6 if the relay is to switch on when the temperature rises above the preset.

Final assembly

Fig.5 shows the front and rear panel artworks. You can download the artworks in PDF format from the *EPE* website.

Mounting the panel meter

The LCD panel meter is mounted by sliding into its front-panel slot (which is open at the top). Check that the top of the meter sits flush with the top of the panel. If it protrudes slightly, it will be necessary to make the slot slightly deeper until it does sit flush.

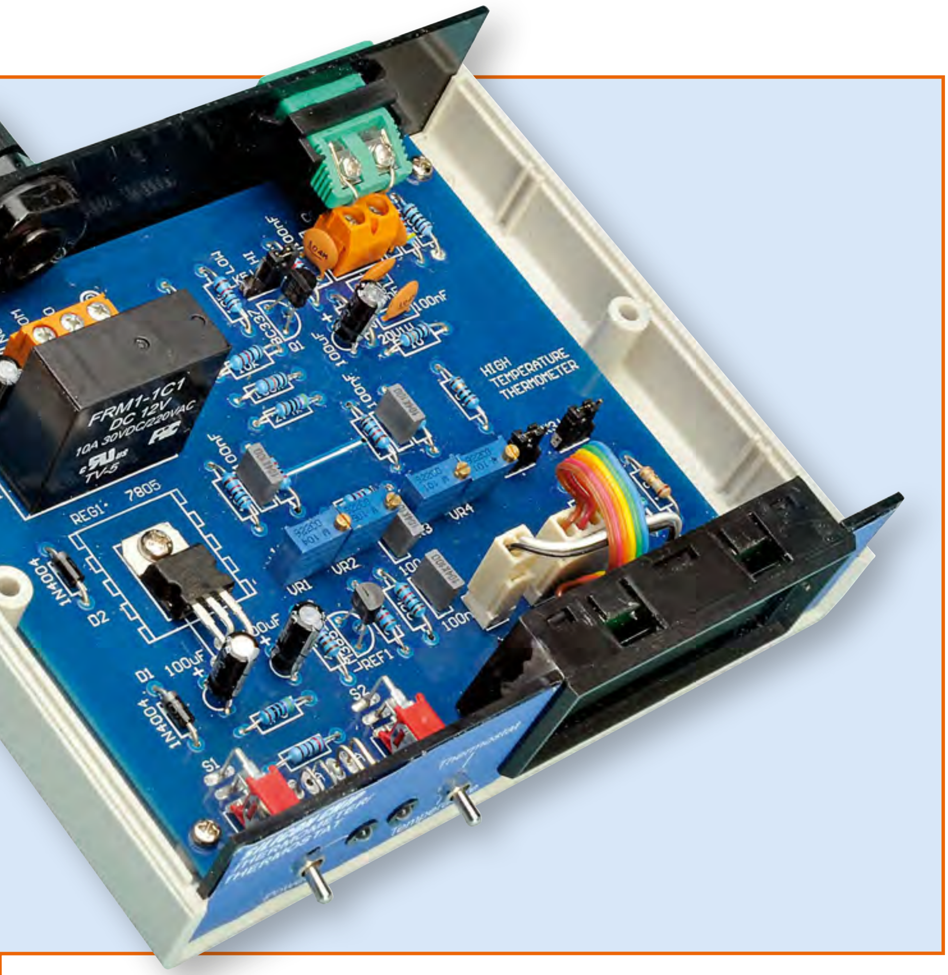
The meter is secured in place by running a bead of silicone sealant or hot-melt glue along the two vertical inside edges, adjacent to the front panel.

Once the meter is in place, the front panel and the PCB assembly can be slid into the case. The PCB is then secured to the base using four self-tapping screws that go into integral mounting bushes. That done, the leads from the panel meter can be plugged into the headers on the PCB.

The rear panel carries a cable gland (for the relay outputs) and the thermocouple socket. The latter is fed through its mounting slot from the rear (terminal screws facing up) and fitted with the supplied clip to hold it in place.

Once that's done, the rear panel can be slipped into the case and two short wires run between the thermocouple socket and the screw terminal block on the PCB.

The lid can now be test fitted to make sure everything is correct. Note that it will be necessary to file the two ridges



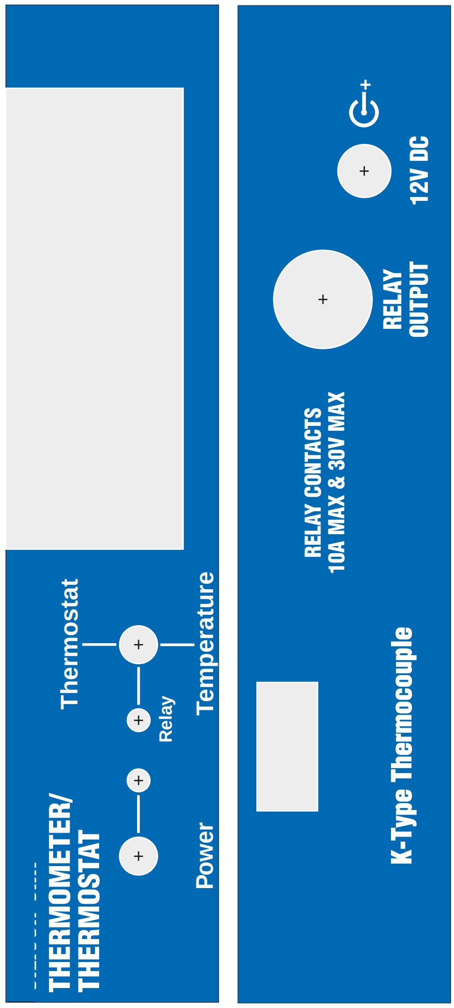


Fig.5: these front and rear panel artworks can be copied and used as drilling templates.

at the front of the lid down where they meet the panel meter.

Testing

To test the unit, first apply power and check the power LED lights. The display should also show a temperature reading with S2 (*Thermostat/Thermometer*) in position 1 (Temperature).

If it does, check the power supply voltages on the board. REG1's output should be close to +5V, while pin 7 of IC1 should be about 11.4V, as should pin 4 of IC2. REF1 should have close to 2.5V across terminals 1 and 2.

Now check that the display shows a temperature that's close to the ambient when the connected probe is exposed to room air. Assuming it does, switch S2 to position 2 (Thermostat) and check that you can adjust the preset using VR1. The relay should click on and off when the preset goes just over or under the measured temperature.

Table 1: Resistor Colour Codes

❏	No.	Value	4-Band Code (1%)	5-Band Code (1%)
❏	1	100kΩ	brown black yellow brown	brown black black orange brown
❏	1	51kΩ	green brown orange brown	green brown black red brown
❏	2	47kΩ	yellow violet orange brown	yellow violet black red brown
❏	1	20kΩ	red black orange brown	red black black red brown
❏	1	15kΩ	brown green orange brown	brown green black red brown
❏	8	10kΩ	brown black orange brown	brown black black red brown
❏	1	4.7kΩ	yellow violet red brown	yellow violet black brown brown
❏	1	2.2kΩ	red red red brown	red red black brown brown
❏	1	1kΩ	brown black red brown	brown black black brown brown
❏	2	470Ω	yellow violet brown brown	yellow violet black black brown
❏	1	100Ω	brown black brown brown	brown black black black brown
❏	1	39Ω	orange white black brown	orange white black gold brown
❏	2	10Ω	brown black black brown	brown black black gold brown
❏	1	1.8Ω (5%)	brown grey gold gold	not applicable

Parts List – High Temperature Thermometer/Thermostat

1 PCB, code 21105121, available from the *EPE PCB Service*, size, 117mm × 102mm
1 plastic instrument case, 140mm × 110mm × 35mm
1 12V DC 500mA plugpack
1 3½-digit LCD panel meter
1 front-panel label
1 rear-panel label
1 K-type thermocouple probe
1 K-type thermocouple probe socket
1 SPDT 10A 12V relay, (RELAY1)
2 SPDT PCB-mount toggle switches (S1,S2)
1 PCB-mount 2.5mm DC socket (CON1)
1 2-way PCB-mount screw terminal block, 5.08mm spacing (CON2)
1 3-way PCB-mount screw terminal block, 5.08mm spacing (CON3)
1 cable gland for 3-6.5mm diameter cable
1 2-way polarised pin header, 2.54mm spacing
1 6-way polarised pin header, 2.54mm spacing
1 2-way header sockets to match above header
1 6-way header sockets to match above header
2 3mm LED bezels (optional)
3 3-way pin headers, 2.54mm spacing (LK1 to LK6)
3 jumper shunts
4 No.4 × 6mm self-tapping screws
1 M3 × 6mm pan-head machine screw
1 M3 nut
1 100mm length of 0.8mm tinned copper wire
1 50mm length of 8-way ribbon cable

Semiconductors
1 AD8495ARMZ precision thermocouple amplifier with cold junction compensation (IC1) (Element14 Cat. 186-4707)
1 OP747ARZ quad precision single supply op amp (IC2) (Element14 Cat. 960-4405) (IC2)
1 LM285Z/LP-2.5 micropower voltage reference diode (REF1) (Element14 Cat. 966-5447; Jaycar ZV1626)
1 7805 5V 3-terminal regulator (REG1)
2 BC337 NPN transistors (Q1,Q2)
1 22V 1W Zener diode (ZD1)
2 1N4004 1A diodes (D1,D2)
1 green 3mm LED (LED1)
1 red 3mm LED (LED2)

Capacitors
3 100µF 16V PC electrolytic
1 10µF 50V non-polarised electrolytic
4 100nF ceramic
4 100nF MKT polyester

Trimpots
1 1MΩ top-adjust multi-turn trimpot (code 105) (VR2)
1 100kΩ top-adjust multi-turn trimpot (code 104) (VR1)
2 100Ω top-adjust multi-turn trimpots (code 100) (VR3,VR4)

Resistors (0.25W, 1%)

1 100kΩ	1 2.2kΩ
1 51kΩ	1 1kΩ
2 47kΩ	2 470Ω
1 20kΩ	1 100Ω
1 15kΩ	1 39Ω
8 10kΩ	2 10Ω
1 4.7kΩ	1 1.8Ω 5%

Reproduced by arrangement with SILICON CHIP magazine 2013. www.siliconchip.com.au

VR2 can now be adjusted to give the required amount of hysteresis (clockwise for more hysteresis and anticlockwise for less).

Calibration

If you wish, the unit can be left uncalibrated, in which case its accuracy will be as shown in the specifications panel.

Alternatively, if you wish to calibrate the unit for improved accuracy, the procedure is as follows:

1) Remove jumper links LK2/LK4 and install links LK1/LK3 instead
2) Place the thermocouple probe in a cup of distilled water brimming with ice (note: the ice also needs to be made from distilled water to ensure accuracy and the ice-water mixture has to be constantly stirred to maintain a 0°C temperature)
3) Adjust VR3 so that the thermometer reads 0°C
4) Place the thermocouple probe in boiling distilled water and adjust VR4 for a reading of 100°C at sea level or deduct 1°C for every 300m above sea level.

The unit can be used with any K-type thermocouple

That completes the calibration. The lid can now be attached to the case and the unit is ready for use.

Ambient temperature display

There are a couple of options available if you just want the unit to measure the ambient temperature.

First, you can use the thermocouple as the sensor and simply sit it in free air. Alternatively, you can disconnect the thermocouple and short its inputs on the PCB using a short length of wire. The unit will then display the ambient temperature (in °C) as measured by the AD8495 itself.

Note that this will really be the temperature inside the case rather than the room temperature. However, this will be close to room temperature, since there is little warming inside the case.

If you intend using this project simply as an ambient temperature thermometer or to measure temperatures up to 199°C only, then the divider resistors can be changed so that they divide by five instead of 50. That way, the display can show the temperature with a 0.1°C resolution.

To do this, change the 1kΩ resistor to 12kΩ, the 39Ω resistor to 750Ω and the 1.8Ω resistor to a 0Ω resistor (or wire link). The 100Ω trimpot (VR4) on the

adjustable side of the divider should be changed to 1kΩ.

Finally, the decimal point in front of the righthand digit can be displayed by connecting the LCD panel meter's DP3 pin to the +5V supply. The details are shown on the instruction sheet supplied with the meter.

Controlling mains voltages

As presented in the diagram and photos, the *Digital Thermometer/Thermostat* is capable of controlling external loads running at 30V DC and up to 10A. However, it can control 230V AC loads, provided the relay and the wiring itself is rated for 250V AC mains operation. **This will mean that a larger case must be used to accommodate the extra wiring and mains input and output sockets** (note: the plastic case used here is not suitable; it's too small and the back is too flimsy to safely anchor mains cables).

The mains input wiring will need to include a mains fuse and we suggest an IEC chassis-mount male socket that includes a switch and fuse. For the output mains wiring, use a chassis or panel-mount female IEC socket or 3-pin mains panel-mount socket.

All mains wiring should be run in 250V AC 10A-rated cabling. Cable tie and clamp the internal mains wires so they cannot possibly come adrift and contact any low-voltage section of the circuit. It's a good idea to secure the terminal block wires to the PCB; eg, by using silicone sealant or a cable tie that loops through a couple of holes drilled through the PCB adjacent to the terminal block.

A metal enclosure will need to be securely earthed. For a plastic case, any exposed metal screws used to secure the IEC connector or other parts near to the mains wiring will also need to be earthed. Nylon screws can be used as an alternative to earthing the screws.

The relay should have contacts rated for 250V AC operation. Finally, for the 3-way terminal block, CON3, we recommend using a Weidmuller type so that it has sufficient voltage rating.

Computer error: reliable digital processing – Part 3

The Curiosity rover lands on Mars with no human intervention (JPL-NASA)

A short history of high-reliability computing by Dr William Marshall

Functional safety and international standards

In the 21st century, nobody likes the idea of taking part in any enterprise that might involve even a small risk to their safety. As far back as the 19th century, railway signalling systems were said to be 'fail-safe' and that phrase has been bandied about ever since. Unfortunately, it has never been true: all a system's designer can do is reduce the probability of catastrophic failure to some small figure, acceptable to, but rarely understood by the travelling public. We are continually learning lessons from disasters, but somehow they keep on happening.

Making sure that a computer system in a vehicle or industrial process, for example a nuclear power station, is as reliable as possible involves a lot more than some clever hardware. The whole process of hardware and software design must be rigorous from start to finish, as must pre- and post-delivery testing. These design processes are formalised in international standards such as IEC61508 for industrial systems and ISO26262 for the specific area of automotive control. We certainly need this design rigour with the prospect of driverless cars on our roads just over the horizon.

Reconciling imperfect components with 'Failure is not an option'

In Part 1, we saw the replacement of thermionic valves by solid-state devices – transistors – and the beginnings of mobile computing driven by the needs of the Apollo moon landing project. Gene Kranz, flight director of the ill-fated Apollo 13 moon mission is quoted as saying to his team, 'Failure is not an option', as they endeavoured to solve the problem of a crippled spacecraft and stranded astronauts. We all know how it turned out and the episode serves to illustrate something that humans are very good at, but machines are not: we have intelligence and can deal with situations when they go wrong. Computers just tend to crash.

The principle of fault avoidance through careful design and rigorous testing did not prevent the failure of a cheap, insignificant component from, at the very least, terminating a very expensive mission. The mission did not end in disaster with loss of life because the flight controllers, engineers and the astronauts themselves provided some 'fault tolerance' at 'run-time'. The astronauts detected the fault, personnel at mission control diagnosed the likely consequences and back-room engineers came up with the workarounds.

Fault-tolerant design

Fault tolerance assumes that faults are likely to occur no matter how many steps are taken to avoid them.

- 'Fault-tolerant' systems have built-in capability (without external assistance) to preserve the continued correct execution of their programs and input/output functions in the presence of a certain set of operational faults.
- 'Fail-safe' systems cannot resume safe operation after fault detection, but will shut down in a predictable manner without producing erroneous outputs.

These deceptively simple definitions are, in fact, very difficult to translate into real systems. Assuming that we have a permanent or transient fault in the system, then three requirements must be met to satisfy the desire for 'continued correct execution'; these are:

1. **Error detection –** the system must be able to detect its own mistakes.
2. **Fault diagnosis –** having detected an error while running the application program, the system must be able to isolate the fault to a group of components or modules, which can be by-passed, replaced under processor control or shut down.
3. **Fault recovery –** once the fault has been located, the system must take action to eliminate or minimise its effect. For a transient fault, a simple 're-try' may be enough.

As we have seen from Apollo 13, early life-critical systems tended to rely on human intervention for all three functions.

Fault tolerance and fault avoidance are not mutually exclusive and both techniques can be combined when working up a particular design. The introduction of redundant components and the inclusion of spare modules do not automatically improve system reliability. Indeed, replication of basically low quality components will make the redundant system less likely to complete its mission than the simple one.

The use of top-quality components and a de-rated design are necessary for the maximum benefit of fault-tolerant computing to be achieved. This means that a system will have increased 'availability' and the probability of mission success is enhanced even with the presence of failed components. For example, special attention to availability issues at the design stage has enabled the Mars Exploration Rovers, launched in 2003, to exceed its expected mission lifetime by many years.

Redundancy

The key to fault tolerance is 'redundancy', whereby extra circuits improve system reliability but don't increase its functional performance. A 'watch dog' timer, sometimes included on the processor chip or as part of a separate supervisor device, is widely used to detect processor failure. It usually forces a system reset when a program-generated signal disappears. These very simple devices often incorporate power supply monitoring as well. Newer microcontroller designs feature on-chip error detection and diagnostic circuits; for example, the Texas Instruments TMS470M. Even the Microchip PIC24/dsPIC devices have a large set of detectable error conditions leading to program interrupts, by

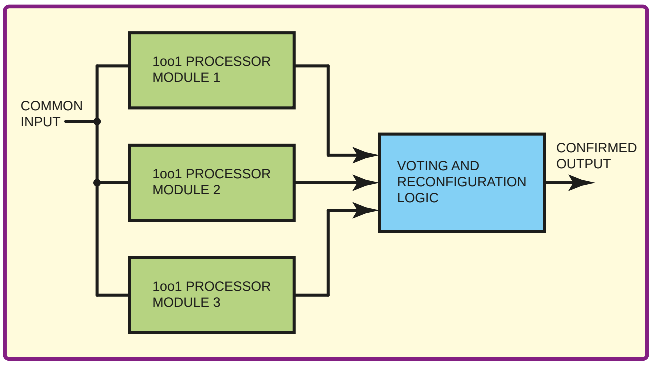


Fig.1. A single-fault tolerant system based on 'triple modular redundancy'

default a reset. However, in order to meet the needs of the new safety standards a much more comprehensive solution is required.

Modern dual-core and quad-core microprocessors used in PCs and smartphones are designed to improve overall performance by introducing parallel processing – each core runs a different program in parallel with the others. However, this is not redundancy. The very latest 'safety microcontrollers' also feature multiple cores, but in their case each runs a copy of the same program. The outputs from each core are compared at intervals. This type of device will actually have lower performance than a single-core unit, but it can at least detect transient or permanent faults and avoid crashing. Such systems have 'dual modular redundancy' (DMR), they are not fault tolerant because they can't tell which processor is at fault; so, in the event of a permanent difference being detected, system shutdown is inevitable. These systems are deemed to be 'fail-safe'. A triple modular redundancy (TMR) system can tolerate one failed processor that's been out-voted by the other two (see Fig.1).

The Texas Instruments Hercules TMS570 microcontrollers contain two processor cores, which execute the same program in 'lock-step', but one serves only as a slave checking device producing outputs for comparison with the master. Lock-step refers to the processor clocks being locked together but not synchronised. A transient error 'event' will not disrupt both processors because one is running a couple of clock cycles behind the other, see Fig.2. Unfortunately, only the master outputs are available to the rest of the system, so DMR cannot be achieved with one device. Still, the redundant core does improve the detectability of faults.

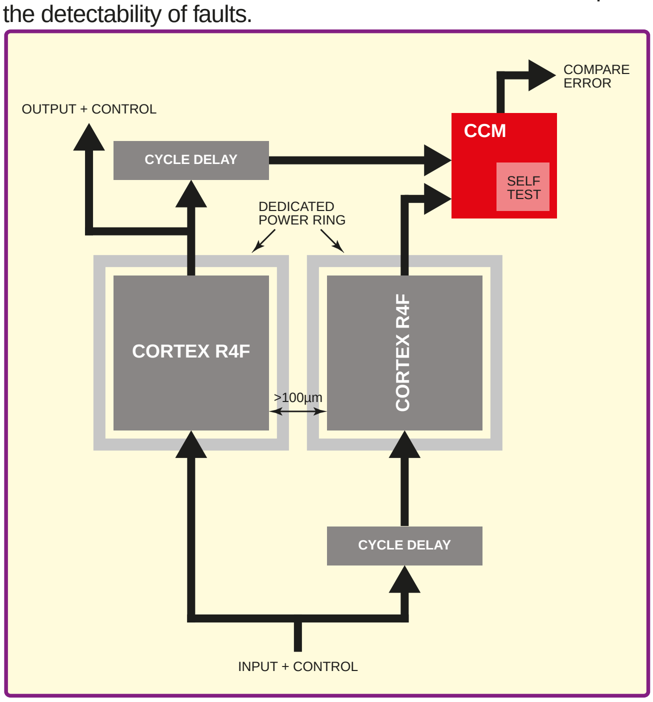


Fig.2. Internal architecture of a Texas Instruments Hercules TMS570 microcontroller

Transient and hard faults

If a checking system comes up with an error, it may just be a one-off caused by the impact of a stray cosmic particle flipping the state of a RAM cell for example. The effect of this transient fault may be eliminated by a simple re-try of the program segment that led to the error. The ability to perform a re-try must be built into the system, otherwise hardware resources will be shut down unnecessarily. Time and effort spent getting these circuits/software right will pay handsome dividends if the system works in an electrically noisy environment. Of course, the error checking system must also be able to sense a 'hard' fault quickly and avoid pointless re-tries.

Static and dynamic redundancy

Basic modular redundancy with voting circuits is normally classified as 'static', where all modules are 'hot' and running. A processor module may be ignored or powered-down when it develops a hard fault. 'Dynamic' redundancy involves hot or cold standby spare units, which are switched in and out as required by fault detection logic and/or software. Dynamic redundancy has been used extensively on the Space Shuttle, Airbus aircraft and the Mars planetary rovers. In the Airbus example, a further precaution was taken against common-mode faults by introducing 'diversity', whereby processor modules are based on different microcontroller platforms with software written by independent teams.

These systems feature dual-processor chip modules, which could now be replaced with single chips, such as the Hercules dual-core devices. For example, two chips could be combined to form a fault tolerant 1oo2D system (see Terminology box and Fig.3). In this case, both processors are 'hot' and both receive the same inputs including a common reset. When a swap is commanded, the outputs of the standby unit replace those of the failed module.

Terminology

1oo1	Basic, non-redundant: 'One out of One' processors must work, not fault tolerant
1oo1D	Basic + Diagnostics: 'One out of One' processors must work, not fault tolerant
1oo2	Dual redundant (DMR): 'One out of Two' processors must work, not fault tolerant
1oo2D	Dual redundant + Diagnostics: 'One out of Two' processors must work, fault tolerant
2oo3	Triple redundant (TMR): 'Two out of Three' processors must work, fault tolerant

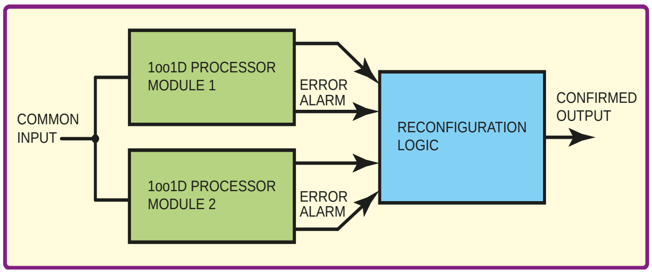


Fig.3. A single-fault tolerant system based on two dual-core safety microcontrollers

Looking back to the future

Finally, we go almost full circle: in order to incorporate more fault avoidance in our redundant design we don't use the latest super-powerful processor chips. In space applications, where ionising radiation can be at very high levels, designers prefer to use old cores that have been around for 20 years or more. For example the Mars Curiosity rover has two RAD 750 computers on board. Each costs about $200,000 and is based on a chip derived from the PowerPC 750 processor that powered Apple computers in the late 1990s!

This is done for two main reasons. First, the chip architecture is fully debugged and all quirks documented. A lot of software expertise is available too. Second, the less dense silicon technology is not so susceptible to corruption by cosmic particle impacts. Cosmic radiation can even play havoc with large servers on the ground because individual elements on each chip are now so small. Memory error correction coding (ECC) is essential to enable these installations to function. Out of the consumer gaze perhaps, but high-reliability technology is working to keep you safe, especially when the time comes to release the steering wheel and let the car drive itself.

EPE PIC RESOURCES CD-ROM V2

ONLY £14.75
INCLUDING VAT and P&P

Version 2 includes the EPE PIC Tutorial V2 series of Supplements (EPE April, May, June 2003)

The CD-ROM contains the following Tutorial-related software and texts:

- EPE PIC Tutorial V2 complete series of articles plus demonstration software, John Becker, April, May, June '03
- PIC Toolkit Mk3 (TK3 hardware construction details), John Becker, Oct '01
- PIC Toolkit TK3 for Windows (software details), John Becker, Nov '01

Plus these useful texts to help you get the most out of your PIC programming:

- How to Use Intelligent LCDs, Julyan Ilett, Feb/Mar '97
- PIC16F87x Microcontrollers (Review), John Becker, April '99
- PIC16F87x Mini Tutorial, John Becker, Oct '99
- Using PICs and Keypads, John Becker, Jan '01
- How to Use Graphics LCDs with PICs, John Becker, Feb '01
- PIC16F87x Extended Memory (how to use it), John Becker, June '01
- PIC to Printer Interfacing (dot-matrix), John Becker, July '01
- PIC Magick Musick (use of 40kHz transducers), John Becker, Jan '02
- Programming PIC Interrupts, Malcolm Wiles, Mar/Apr '02
- Using the PIC's PCLATH Command, John Waller, July '02
- EPE StyloPIC (precision tuning musical notes), John Becker, July '02
- Using Square Roots with PICs, Peter Hemsley, Aug '02
- Using TK3 with Windows XP and 2000, Mark Jones, Oct '02
- PIC Macros and Computed GOTOs, Malcolm Wiles, Jan '03
- Asynchronous Serial Communications (RS-232), John Waller, unpublished
- Using I2C Facilities in the PIC16F877, John Waller, unpublished
- Using Serial EEPROMs, Gary Moulton, unpublished
- Additional text for EPE PIC Tutorial V2, John Becker, unpublished

NOTE: The PDF files on this CD-ROM are suitable to use on any PC with a CD-ROM drive. They require Adobe Acrobat Reader – included on the CD-ROM

EPE PIC RESOURCES V2 CD-ROM ORDER FORM

Please send me .... (quantity)

EPE PIC RESOURCES V2 CD-ROM

Price £14.75 each – includes postage to anywhere in the world.

Name ....

Address ....

....

....

.... Post Code ....

☐ I enclose cheque/P.O./bank draft to the value of £ ....

☐ please charge my Visa/Mastercard/Maestro £ ....

Card No. ....

Card Security Code ....
(The last 3 digits on or just under the signature strip)

Valid From .... Expiry Date ....

Maestro Issue No. ....

SEND TO:
Everyday Practical Electronics,
Wimborne Publishing Ltd.,
113 Lynwood Drive, Merley, Wimborne,
Dorset BH21 1UU.
Tel: 01202 880299. Fax: 01202 843233.
Email: orders@epemag.wimborne.co.uk

Payments must be by card or in £ Sterling – cheque or bank draft drawn on a UK bank.

Normally posted within seven days of receipt of order. Send a copy of this form, or order by letter if you do not wish to cut your issue.

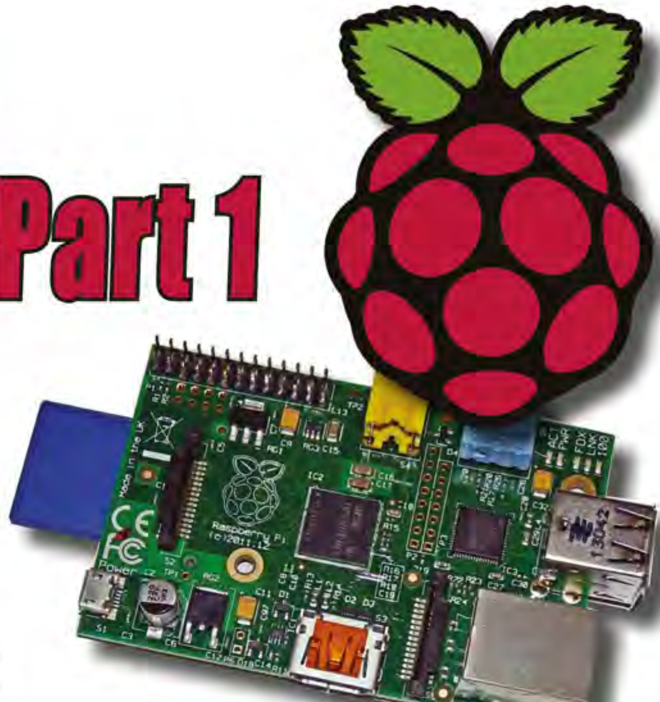
***Order on-line from* www.epemag.com or by Phone, Fax, Email or Post.**

BECOME A PIC WIZARD WITH THE HELP OF EPE!


FREE CD-ROM
ELECTRONICS TEACH-IN 5
£8.99
FROM THE PUBLISHERS OF EPE EVERYDAY PRACTICAL ELECTRONICS
JUMP START
15 design and build circuit projects dedicated to newcomers or those following courses in schools and colleges
PRACTICALLY SPEAKING
The techniques of project construction
PIC 'N MIX
Starting out with PIC microcontrollers
FREE CD-ROM
TEACH-IN 2
TWO TEACH-INs FOR THE PRICE OF ONE!
The free CD-ROM provides a practical introduction to PIC microcontrollers
Plus MikroElektronika, Microchip and L-Tek PoScope software
ISBN 978-1-898805-23-6
ONLY £8.99!

Teach-In 2014
Raspberry Pi – Part 1

by Mike and Richard Tooley



Welcome to *Teach-In 2014* with Raspberry Pi. This exciting new series has been designed for electronics enthusiasts wanting to get to grips with the immensely popular Raspberry Pi, as well as computer buffs eager to explore hardware and interfacing. So, whether you are considering what to do with your Pi, or maybe have an idea for a project but don't know how to turn it into reality, our new *Teach-In* series will provide you with a one-stop source of ideas and practical information.

The Raspberry Pi offers you a remarkably effective platform for developing a huge variety of projects; from operating a few lights to remotely controlling a robotic vehicle through the Internet. *Teach-In 2014* is based around a series of practical exercises with plenty of information for you to customise each project to meet your own requirements.

The Raspberry Pi is no mean performer; it can offer you very similar performance to that which you might expect from a larger and much more expensive computer system, so don't be fooled by the relatively small price tag. By shopping around you can build a very effective computer system based on a Raspberry Pi for less than £100. However, if you are looking for something more modest and just want to take advantage of the Raspberry Pi as a single-board computer for a particular control application then you can be up and running for a very reasonable outlay.

This series will teach you about:

- **Programming** – introducing you to the powerful Python programming language and allowing you to develop your programming skills
- **Hardware** – learning about the components and circuits that are used to interface microcomputers to the real world
- **Computers** – letting you get to grips with computer hardware and software and helping you understand how they work together
- **Communications** – showing you how to connect your Raspberry Pi to a network and control a remote device using Wi-Fi and the Internet.

So, what's coming up? Regular features of *Teach-In 2014* with Raspberry Pi will include:

- **Pi Project** – the main topic for each part will be a project that explores a particular use or application of the Raspberry Pi in the real word. Projects will include shopping for your Pi, set up, environmental monitoring, data logging, automation and remote control.
- **Pi Class** – each of our Pi Projects will be linked to one or more specific learning aims. Examples will include methods of representing and handling data, serial versus parallel data transmission and architecture of a microprocessor system.
- **Python Quickstart** – a short feature devoted to specific programming topics, such as data types and structures, processing user input, creating graphical dialogues and buttons and importing Python modules. We will help you get up and running with Python in the shortest time!
- **Pi World** – this is where we take a look at a wide range of Raspberry Pi accessories, including breadboards, prototype cards, bus extenders and Wi-Fi adapters. We will also help you build your Raspberry Pi bookshelf with a selection of recommended books and other publications.
- **Home baking** – suggested follow-up and extension activities such as 'check this out', a simple quiz, things to try and websites to visit.
- **Special features** – an occasional 'special feature'. For example, how to laser cut your own mounting plate – with additional downloadable resources such as templates and diagrams.

What will I need?

To get the best out of our series you will, of course, need access to a Raspberry Pi. If you don't already have one, don't worry – we will be explaining what you need and why you need it (we will also be showing you how you can emulate a Raspberry Pi using a Windows PC).

This month

This month, we will tell you what hardware and software you will need to set up a Raspberry Pi system. In *Pi Project* we will explain what you need to get started, complete with a shopping list of the items that you might need. Our *Pi Class* sets the scene with a brief tour of the Raspberry Pi's hardware, before explaining what a microcomputer system is and how the hardware and operating system software work together to provide you with a complete and functioning computer system. *Home Baking*, our practical feature for constructors and software developers, will take you through the process of installing and configuring the Raspberry Pi's operating system. Finally, *Python Quickstart* (our section on programming) will provide you with an introduction to the Python programming language and its integrated development environment.

Pi Project

In this month's *Pi Project* we will be looking at what you need in order to build a complete computer system based on a Raspberry Pi. If you already have a Raspberry Pi set up and working you can ignore this and move on to our *Pi Class* on page 43.

The Pi Shop

OK, so you want to get into Raspberry Pi, but you're not sure what you need to buy to get started. This guide will help you to identify what you need, along with some practical buying advice.

The first thing you'll discover is that although you may have heard about the amazing '£30 computer system', in truth, there are a few extra bits and pieces that you'll need before you can get started. In fact, 'in the box' all you'll find (apart from the obligatory anti-static bag) is the Pi motherboard itself. The good news is that the extra bits you'll need should cost you no more than £30 to £40, so it still represents great value for money, considering the fun that we're sure you'll get out of hours of tinkering! Whatever you do, don't suffer the frustration of just ordering the Pi, waiting eagerly for it to arrive and then finding that you can't get up and running straight away.

Next, we'll take a look at what you need, as well as giving you some practical buying advice.

Many of the major resellers and independent suppliers have cottoned on to this problem and offer starter 'bundles' that contain everything that you need to get going, including a pre-loaded operating system (see *Pi Class*). These packages often represent a significant discount on buying individual items, but if you're like the authors, you may already have some of the hardware items that you'll need and might only need to buy a few extra items in order to build a complete system. Alternatively, you may wish to shop around for items to get the best price or stick to particular brands that you like. The big advantage with buying a bundle is the convenience of getting everything in one go and the guarantee that each part will be compatible. Whatever avenue you decide to follow, here's what you'll need:

Fig.1.2. An HDMI-to-VGA adapter

Major Pi accessories

Raspberry Pi

Obviously, your number one purchase is the Raspberry Pi main board itself. The Raspberry Pi is widely available from large electronic suppliers (such as CPC and RS), small hobbyist retailers and individual sellers on eBay. Your first choice will be which model of Pi to purchase. There are two versions – Model A and Model B. Model B (see Fig.1.1) should set you back about £30 and has 512Mb of memory, on-board Ethernet and two USB ports. The cheaper Model A version has the same 700MHz ARM processor, but less memory (256Mb), no Ethernet and only a single USB port. A Model A board will only set you back around £20. Which version you choose will depend on your intended use. However, if it's your first adventure into the world of Raspberry Pi and you're interested in trying out what it can do for you, then we'd definitely advise going for a Model B. (We'll be making use of the Model B version throughout the series.) We'll tell you more about the two versions and their differences in a later *Teach-In 2014* article.

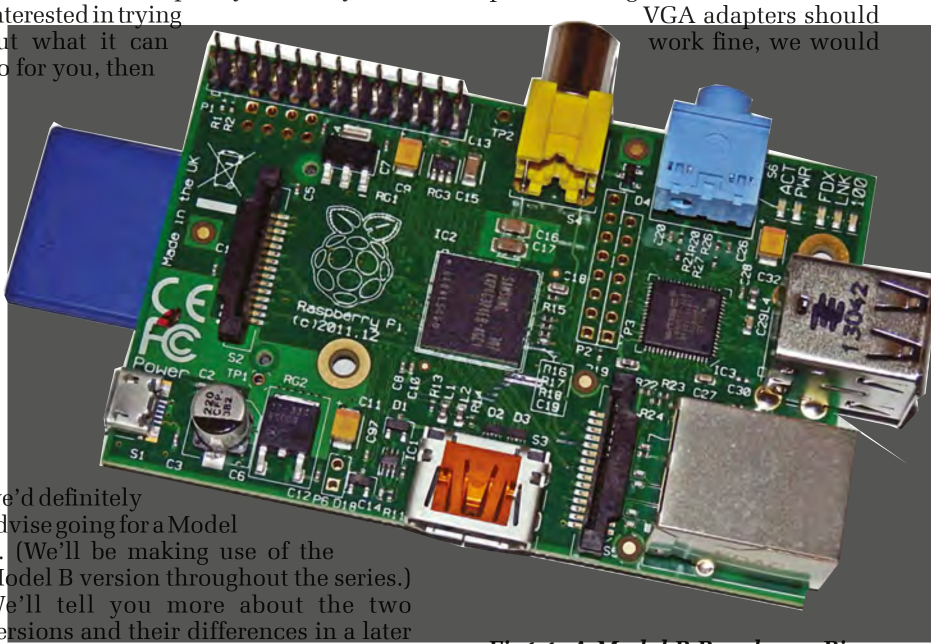


Fig1.1. A Model B Raspberry Pi

Power supply

The Pi operates from a standard 5V micro-USB power source. The Pi does have a relatively high current requirement (something that has been noted by critics) and a power supply capable of delivering a minimum of 1A is recommended. Remember that the power supply will also be used for any output devices, so take this into account when estimating your individual requirements. If you are planning on a 'portable' application, there are various USB rechargeable battery packs on the market that could be used.

We would recommend avoiding low cost, poor quality chargers available from online auction sites for as little as 99p. They are often very cheaply manufactured and poorly designed, resulting in a 'dirty' output, electromagnetic interference, overheating and safety concerns. Similarly, many chargers are designed for mobile phone charging and are not rated sufficiently for extended periods of use. Our advice is simple – spend a little more and get a Pi-specific charger from a reputable source.

HDMI lead or VGA converter

One choice that you'll need to make is what type of display you are going to use with your Pi. The Pi is equipped with a standard HDMI socket, which makes it easy to attach to any modern TV or monitor that supports HDMI using a standard HDMI lead. These are universally available for under £5. However, if you're intending to connect your Pi to a PC monitor with VGA or DVI, you'll need to purchase an appropriate HDMI-to-VGA adapter. These should cost you around £10-20. Generally, they are powered from the HDMI port and provide a standard female VGA connector to which you'll also need to attach a standard VGA lead to connect to your monitor. If your monitor requires DVI, you may also require an additional adapter. Although universal HDMI-to-VGA adapters should work fine, we would recommend one that specifies Pi support to ensure compatibility.

An alternative to using the HDMI output (with or without a VGA adapter) is the composite video RCA output. Generally this would be used to output to an older TV or CRT screen without HDMI capability. Note that the composite output is rather low resolution when compared with the other display methods, and in practice is not ideal for working with the operating system because text can be somewhat unclear. However, it might be a suitable option for multimedia applications running on older display screens.

The Pi is capable of outputting video signals at a range of resolutions/modes up to 1920 × 1200 pixels. You'll need to select a compatible mode before you'll be able to see anything when you boot up. This represents a potentially frustrating stumbling block for new Pi users, who might expect to 'plug and play'. It begs the question, 'if you can't see the screen – how can you set the correct mode?'. The answer is that there's a configuration file on the SD card that can be edited on your PC to set a number of boot options for the Pi, including the graphics mode. Check out our *Home Baking* section later on.

SD card

Other than random access memory (RAM) there's no memory storage natively on the Pi mother board. Therefore, you need to add an SD card to hold your operating system and files. The Pi has a standard SD card slot located on the underside of the board (see Fig.1.3).

Fig.1.3. An SD card fitted to the slot in the underside of the Raspberry Pi board

You've got a couple of options when it comes to your SD card. Many suppliers offer pre-loaded SD cards with an operating system already pre-installed and ready to go out of the box. The advantage of this is you avoid the process of creating your own card image (see *Home Baking* later on). However, be wary of independent sellers who may use poor quality cards and note that as the operating systems are continually in development, your new Pi may arrive with an older version of the OS that will require subsequent updating. You may well pay a premium for a pre-configured SD card compared with a blank card of the same capacity.

If you elect to create your own SD card image you can follow our simple guide (see page 45). The process involves formatting the card, downloading and writing your image. You'll need an SD card of a suitable size – we recommend a minimum of 2Gb for a basic install, preferably larger, particularly if you intend to store multimedia items or other large files alongside the OS. If you're buying a new card to use, then as with any memory card purchase we recommend buying a reputable brand of a high specification to ensure reliability and speed. You should expect to spend less than £10 for a good 8Gb card.

Keyboard/mouse

As with any computer, you'll need to buy input devices to let you point, click, and enter text. Standard USB keyboards and mice should work fine without any additional drivers required. A neat option is to use a wireless keyboard/mouse, as this cuts down on messy wires and reduces the number of USB ports required (particularly with the Type A models that only have a single port). There are also some nice remote-control-style hand-held keyboards available with built-in touch pads, which might represent an attractive option, particularly if you are intending to use the Pi as a multimedia device.

Ethernet lead (optional)

If you're planning on connecting your Raspberry Pi to your home network and/or the Internet, you'll need either a standard Ethernet cable to connect to your home router (Model B only) or you could go wireless and invest in a USB Wi-Fi dongle.

Wi-Fi dongle (optional)

Most of the operating systems designed for the Raspberry Pi have built-in support for standard USB Wi-Fi devices and also offer a software utility to manage wireless network connections in much the sameway as in Windows (see Fig.1.4). However, we would recommend going for one that specifies Raspberry Pi compatibility. Some manufacturers have cottoned on to this and provide specific Raspberry Pi models (such as the 'Wi-Pi'). However, there is generally nothing special about these adapters, other than the guaranteed compatibility, and they often have an inflated price tag. Therefore, our advice is to look for a tried-and-tested model at a reasonable price point. Many sellers specify their compatibility on the listings. Fig.1.5 shows a Dynamode Wi-Fi adapter, available for under £10, and by virtue of its 'nano' form factor, it fits neatly next to our wireless keyboard/mouse transceiver.

Fig.1.4. The Raspbian Wi-Fi config utility

USB hub (optional)

Depending on what you intend to use your Raspberry Pi for, you may need to connect one or more USB devices. Model B boards come equipped with two USB ports, and Model A boards just one. Therefore, you'll easily run out of space to connect peripherals on either model. For example, on a Model B board, if you've connected a wired USB mouse and keyboard you'll have no space left over for a Wi-Fi dongle! Therefore, it's worth investing in a USB hub.

We would avoid un-powered USB hubs because the additional peripherals would then need to draw their current directly from the Raspberry Pi power supply and you could quickly meet or exceed its capacity. Instead, a powered hub should be used. It's worth noting here that as the 5V supply rail for the USB is common with that of the Raspberry Pi, it's possible to power the Pi itself from the hub power supply. This could be desirable or undesirable, but just keep an eye on the load required and the rating of the power supply that you use.

Plug in and play?

Once you've unpacked and connected it all together, you're ready for your first run… well, almost. If you purchased a blank SD card then you'll need to create a suitable boot image before you can get going. Our boot image guide below will take you through this step by step. However, if you purchased a pre-installed SD card then you're a step ahead of the game here, but you may still need to make a few alterations to the **config.txt** file in order to get your Raspberry Pi booting correctly. Many over-eager users are in a hurry to plug everything in and switch on, but are then disappointed when they fail to get a proper boot. In fact, a very common post on Raspberry Pi forums is from first-time users who can't get their first boot. Check out our **config.txt** guide on page 47 to find out more about how to get the initial settings right.

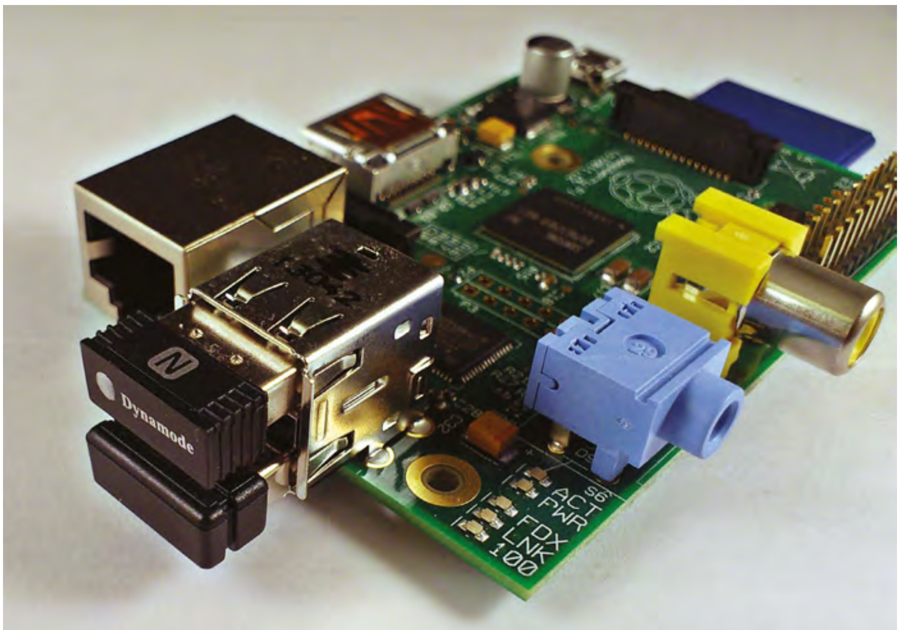


Fig.1.5. Two wireless USB transceivers fitted to a Raspberry Pi

Pi Class

In this month's *Pi Class*, we will look at what goes on inside the Raspberry Pi and how the chips work together to make up a powerful microcomputer system (see Fig.1.6). We start by providing a quick look at the Raspberry Pi's hardware.

Raspberry Pi hardware tour

There's no doubt about it, the Raspberry Pi is an incredible piece of kit with almost limitless applications for electronics and computing enthusiasts alike – and all for less than the price of a pair of jeans. It's no wonder everyone wants one! So what exactly do you get for your £30? Let's take a quick hardware tour.

Computing power

The Raspberry Pi uses a Broadcom ARM-based SoC (system on chip) processor running at a default speed of 700MHz. An integral Videocore 4 video processing unit allows for hardware decoding of high-definition video at high bit rates. To give you an idea of what that means in real terms, Raspberry Pi themselves liken its performance to that of a '300MHz Pentium 2, only with much, much swankier graphics'. However, the Raspberry Pi is not intended to compete as a desktop computer, but its hardware graphics performance and modest ARM processor are perfect for embedded applications and multimedia projects. Although both Models A and B have the same processor hardware, Model B does benefit from its increased 512Mb of RAM compared with the 256Mb fitted to Model A.

General purpose input and output

One of the features that really excites electronic hobbyists is the GPIO (general purpose input output). This allows the connection of all manner of additional inputs and outputs that are accessible from your program running on the Pi. Couple this with the connectivity, networking and processing capabilities and you'll soon discover a really formidable development system capable of supporting some really clever projects. In next month's edition of *Teach-In 2014* we'll be exploring the GPIO and writing our first programs using GPIO inputs and outputs.

Peripheral connectors

Both Model A and B support USB, with one port on A models and two on B. Additional I/O connectors are available on board, allowing the connection of proprietary display and camera modules (we'll be looking at extension modules in a later edition).

Networking connectivity

Model B boards have on-board wired 10-100 Ethernet, and both models are able to use compatible Wi-Fi dongles via USB. This opens up lots of exciting possibilities in terms of connected or web-enabled applications.

Video and audio

As discussed in *Pi Class*, video options include built-in HDMI, providing high-definition video for direct connection to a suitable HDMI monitor or TV. A standard RCA (or 'phono') composite video output is also available if using a legacy display device. HDMI-to-VGA adapters may be used to facilitate connection to a standard monitor. Audio output is available via a stereo 3.5mm jack or directly over HDMI. Audio input is not an on-board feature, but may be achieved via additional compatible USB devices.

Size

The Raspberry Pi has an impressively small form factor of just 86 × 56mm; small enough to be hidden within an embedded system or mounted behind a monitor for example. A range of casing and mounting products are available on the market, and we'll be looking at some of these throughout the series, as well as how to make your own.

Microcomputer systems

Having given you a quick tour of the Raspberry Pi we will next take a look at the components that make up a generic microcomputer system like that shown in Fig.1.7. The central processing unit (CPU) performs three functions: it controls the system's operation; it performs algebraic and logical operations; and it stores information (or data) while it is processing. The CPU works in conjunction with other chips,

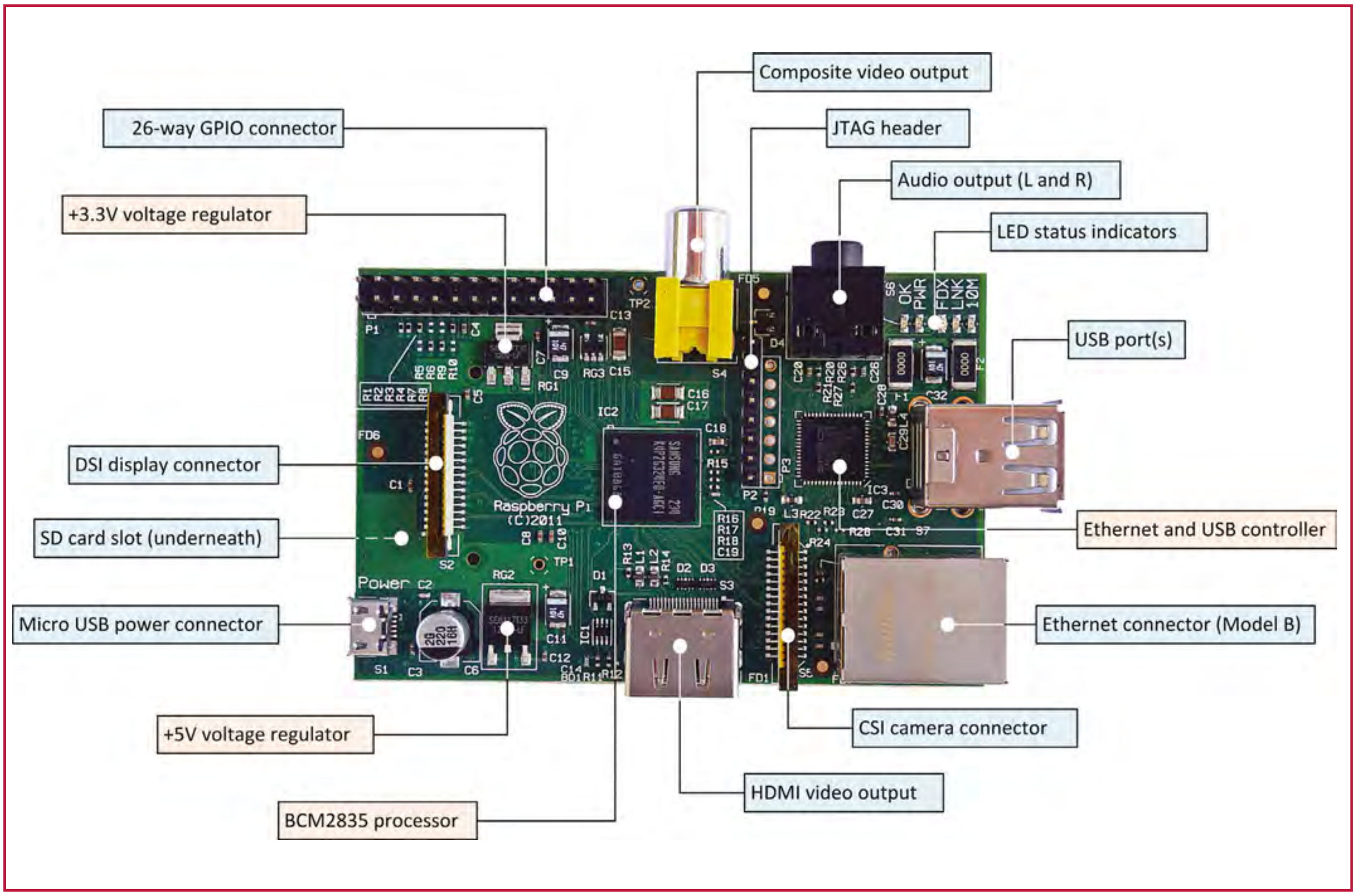


Fig.1.6. The Raspberry Pi is a complete example of a microcomputer system. This diagram shows the main components present on the Pi's circuit board

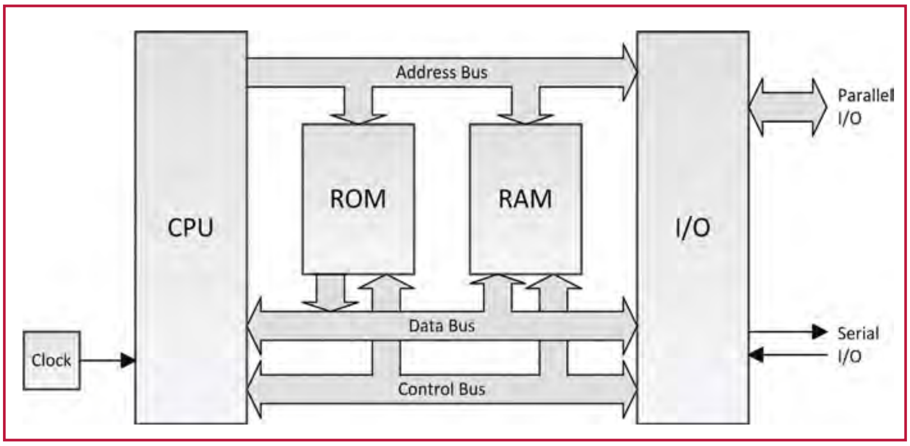


Fig.1.7. Simplified block schematic of a generic microcomputer system

notably those that provide random access memory (RAM), read-only memory (ROM), and input/output (I/O). On the Raspberry Pi, most of these functions are integrated into a single microprocessor chip supported by a handful of other components, including an additional chip with specialised I/O facilities.

The key process in the development of increasingly powerful processor chips has been *microlithography*. In this process, the circuits are designed and laid out using a computer before being photographically reduced to a size where individual circuit lines are about 1/100 the width of a human hair. Early miniaturisation techniques, which were referred to as *large-scale integration* (LSI), resulted in the production of the first generation of 256K-bit memory chips (note that such a chip actually has a storage capacity of 262,144-bits, where each bit is a binary 0 or 1). Today, as a result of *very-large-scale integration* (VLSI), chips can be made that contain the equivalent of several million transistors.

The first microprocessors were developed in the early 1970s. These were simple and crude by today's standards, but they found an immediate application in the automotive industry, where they were deployed in engine management and automatic braking systems and later in the first generation of home computers, such as the immensely popular Sinclair ZX81, Spectrum and Commodore C64.

To make sure that all the data flow within the system is orderly, it is necessary to synchronise all of the data transfers using a *clock* signal. This signal is often generated by a clock circuit (similar to the clock in a digital watch – but much faster). To ensure accuracy and stability, the clock circuit is usually based on a miniature quartz crystal.

All microprocessors require access to read/write memory, in which data (eg, the results of calculations) can be temporarily stored during processing. While some microprocessors (often referred to as *microcontrollers*) contain their own small read/write memory, this is usually provided by means of semiconductor *random access memory* (RAM).

Microprocessors generally also require more permanent storage for their control programs and, where appropriate, operating systems (such as Linux) and high-level language interpreters (such as Python). This is usually provided by means of semiconductor *read-only memory* (ROM). Note that in most modern microprocessor systems, the ROM takes the form of a semiconductor *Flash memory* device that can be updated with a new version of the operating system whenever the need arises.

To fulfil any useful function, a microprocessor system needs to have links with the outside world. These are usually supplied by means of one or more VLSI devices, which may be configured under software control and are therefore said to be programmable. The input/output (I/O) devices fall into two general categories: *parallel* (where a byte is transferred at a time along eight wires), or *serial* (where one bit is transferred after another along a single wire).

The basic components of a microprocessor system (CPU, RAM, ROM, and I/O) are linked together using a multiple connecting arrangement known as a *bus*. The *address bus* is used to specify memory locations (ie, addresses), the *data bus* is used to transfer data between devices, and the *control bus* is used to provide timing and control signals (such as read and write, reset and interrupt) throughout the system.

In addition to the bus lines that link the various system components together, each device needs power. Many small microcomputers need a *supply voltage* of either +5V or +3.3V, but others, like the Raspberry Pi, may also require lower voltages of +2.3V or even +1.8V. These voltages must be obtained from a power supply capable of delivering sufficient current (up to 1A at +5V in the case of a Raspberry Pi) and the voltage must be maintained within fairly close limits (typically ±5%). This means that there needs to be some way of regulating the supply voltage, as well as protecting the microcomputer in the event of a supply malfunction. The Raspberry Pi incorporates no less than four separate on-board voltage regulators and an electronic fuse to provide additional 'last-ditch' protection (see Fig.1.8).

Finally, in order to provide a fully functioning system, we need software capable of managing the various hardware resources, as well as providing us with an interface so that we can interact with the system and execute a series of program instructions written in a human-readable form. This is the function of the *operating system*, together with a variety of utility programs, such as text editors, programming language interpreters, and graphical user interfaces (GUI). You will be getting to grips with all of these in due course.

Fig.1.9 shows how the operating system manages the interface between one or more user application programs and the hardware components of the microprocessor system, together with any peripheral devices (such as keyboards, displays and printers) that are connected. The operating system operates at both 'high level', providing a variety of system services, such as those needed to operate a file system, and at 'low level', using a number of

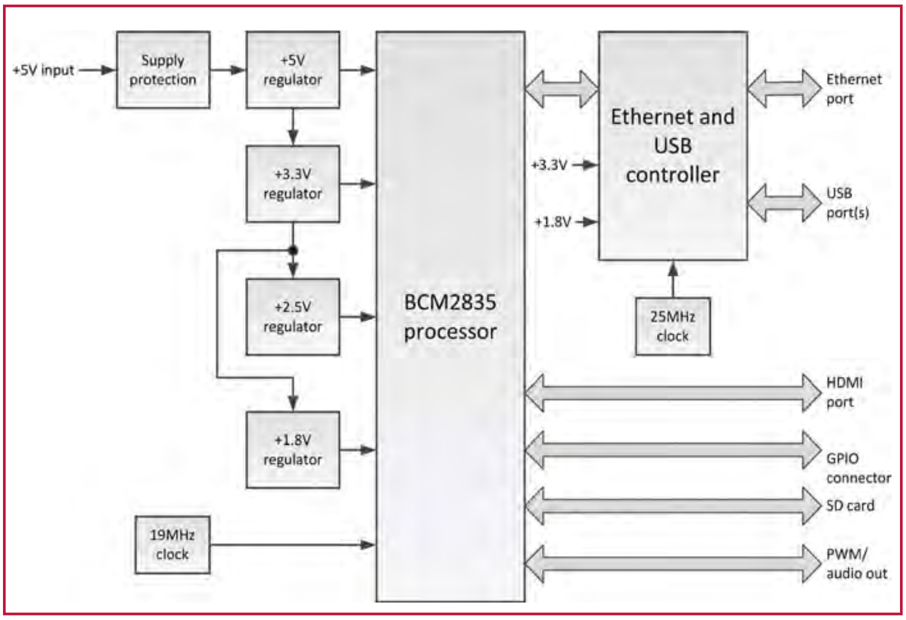


Fig.1.8. Simplified block schematic of a Raspberry Pi showing the two principal chips and voltage regulators

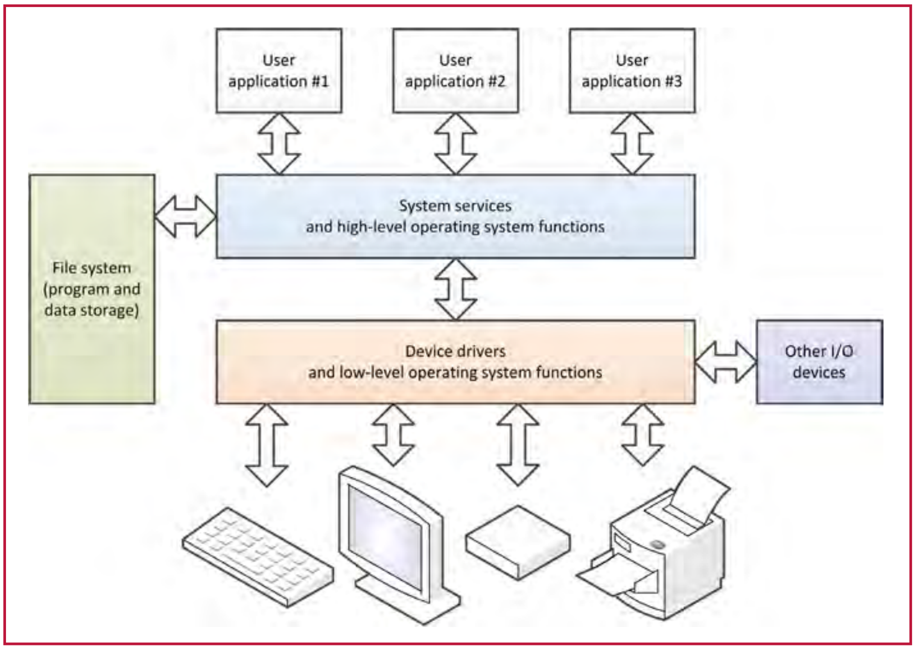


Fig.1.9. How the operating system interacts with and helps to manage the system's resources

drivers that allow the system to interact with specific items of hardware, such as displays and printers.

Home Baking

In our first *Home Baking* feature we will show you how to install and configure the Raspberry Pi's operating system and what to expect when you boot your system for the first time.

Making your SD card image

In this guide we'll be showing you how to create your own SD card image. The SD card on your Raspberry Pi acts like the hard drive on a desktop computer and contains the operating system. An operating system is the background software that runs the hardware and provides the framework to support the applications that run on it. When the Raspberry Pi boots up, it first looks at the SD card for the operating system, which is then executed to start up the system.

There are various operating systems that have been designed to run on the Raspberry Pi; we'll look at them in a later edition of *Teach-In 2014*. The most popular operating system is Raspbian 'wheezy' and is the best one to go for if you're just starting out. Raspbian is a free, open-source operating system based on Debian (a version of Linux). Preparing the SD card is not just a case of dragging and dropping the files into the drive; the SD card must also be formatted and made 'bootable', so that it's ready for operation. Here are the three steps to create your very own Raspbian SD card image:

1. Download and unzip

Visit **www.rasberrypi.org/downloads** and download the latest version of the operating system (in our case Raspbian wheezy). The compressed .zip file is around 500Mb and may take a little time to download on slower Internet connections. You'll also need to download a copy of a utility called W32 Disk Imager that will be used to write the image later on (also linked from the same page). Once downloaded, both .zip files need to be un-zipped.

2. Write the image

Insert your SD card and note the drive letter that is assigned to it. Run Win32DiskImager.exe from the extracted folder.

Once open, click on the folder icon to the right of the first box, select 'Image' and locate the .img (image file) that was extracted from the image .zip file. In the right-hand 'Device' drop-down box, ensure that the drive letter selected is that of your SD card.

When you're ready to start writing, click the 'Write' button; a confirmation message will appear, click 'Yes'. A progress bar tracks the writing progress, followed by a 'write successful' message.

Fig.1.10. The Win32 Disk Imager application

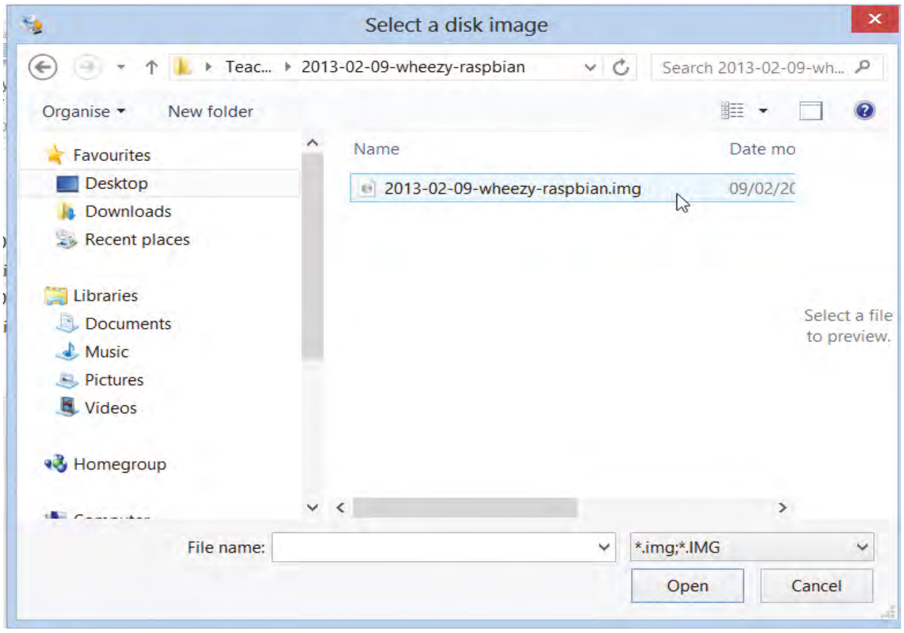


Fig.1.11. Locating the Raspbian image file

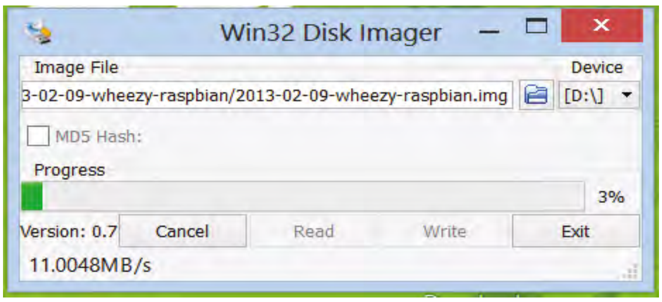


Fig.1.12. Image writing in progress

New Out of Box Software (NOOBS)
The Raspberry Pi community has recently released 'New Out of Box Software' (NOOBS) designed to simplify the 'first run' process. NOOBS is installed on to an SD card (4Gb or larger) in much the same way as described above, but the user is offered a choice of operating systems to install upon boot. Users are also able to change graphics modes using the number keys on the keyboard, hence avoiding manual modification to the **config.txt** file. More information about NOOBS can be found at:
www.raspberrypi.org/downloads

3. Setup for your first run
Your SD card now contains the OS. However, before you take it out of your computer you may need to make a couple of tweaks to **config.txt** to get the setup right before you're ready for your first boot. Check out our guide to **config.txt** for more information.

First boot

The moment has finally arrived to boot your Pi for the first time; everything is plugged in, SD card inserted and settings tweaked, ready to begin. So, what should you expect?

A common question is 'where are the power/reset buttons?' The simple answer is that there aren't any. As soon as power is connected, the Pi will start to boot. This might of course be undesirable. For example, if you're still plugging in your peripherals. Therefore, always plug in the power supply last. A good tip is to plug the power supply into a switched socket so that you can plug everything in before turning power on at the plug. This will also help to prolong the life of the Pi's power socket that can be damaged relatively easily with repeated insertions/removals.

Later, when it comes to turning off the Pi, you should *not* simply cut the power – always execute a proper system shut down (see later).

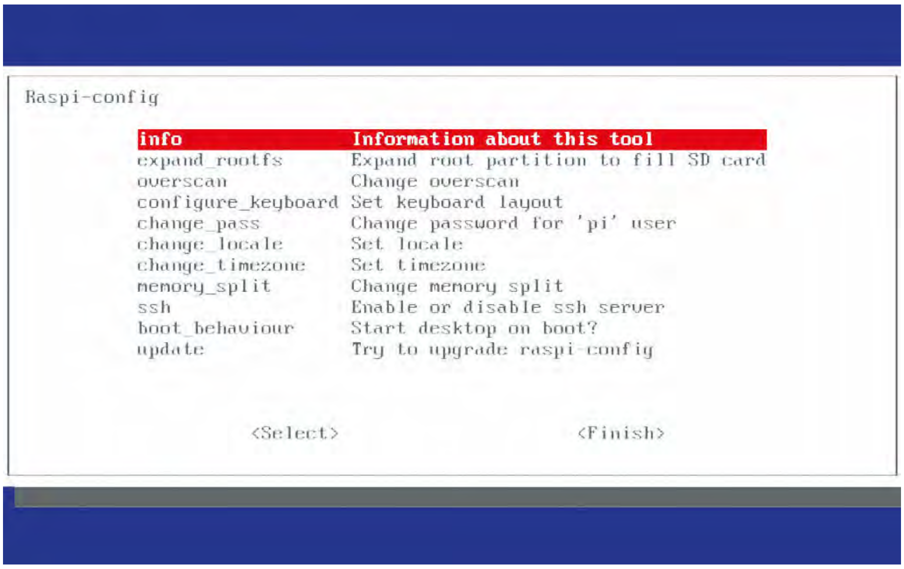


Fig.1.14. The Raspi-config screen

As the Pi boots, you'll see the Raspberry Pi logo, followed by a whole load of text written to the screen. This is quite normal – the Pi loads up the hardware drivers and settings ready to begin.

The first time that you boot the Pi you will be presented with a 'Raspi-config' window. Here you can set up some of the system settings. At this point, it's worth setting the correct time zone and locale. You should also utilise the second option, `expand_rootfs`. Basically, this expands the root partition on your SD card to make use of all of the available space on it.

It's fine to leave the remainder of the settings as default (you can return to this configuration window at a later date if necessary, using the command `raspi-config` from the command line). Once complete select `<Finish>` and reboot the Pi.

Fig.1.15. The Raspberry Pi's command line

Once rebooted, you should be presented with a prompt to login: `raspberrypi login:`. Type pi and hit return; this is the default username. Then, enter the password `raspberry`. This is the default password and can be changed if required in `raspi-config`. Note that as you type the password nothing will appear on the screen – this is normal to protect privacy (you won't see asterisks (stars) appearing, common in password boxes in other systems). You'll then see what is known as a command prompt: `pi@raspberry ~ $`

If you're old enough to remember DOS or you have experience of Linux, you'll be familiar with the command line interface. If you're a younger reader, you're probably familiar with a rich graphical environment like Windows or Apple's OS-X. In the days before computers were capable of providing this type of graphical user interface (GUI) a command line interface was used where words or 'commands' were typed in by the user in order to perform operations like listing files, moving/copying/renaming, running applications and changing settings. The advantage of this was that operations were fast and efficient, with no computing power being 'wasted' on producing the pretty interface. However, it did mean that a user had to remember and know how to use several commands in order to achieve a job.

At this point, we could use the command line interface to work on the Pi. However, Raspbian does have its own Windows-like GUI. To start this you'll need to type the command `startx`, followed by hitting the enter key. A familiar style of GUI will then load, as shown in Fig.1.16.

We'll leave you here to explore the interface and the applications available for yourself. In next month's edition of *Teach-In 2014*, we'll be giving you a full tour of the default applications and features of Raspbian, as well as how to connect to the Internet and add further applications. In the meantime, make yourself at home with the interface and have a good experiment with what's there – have fun!

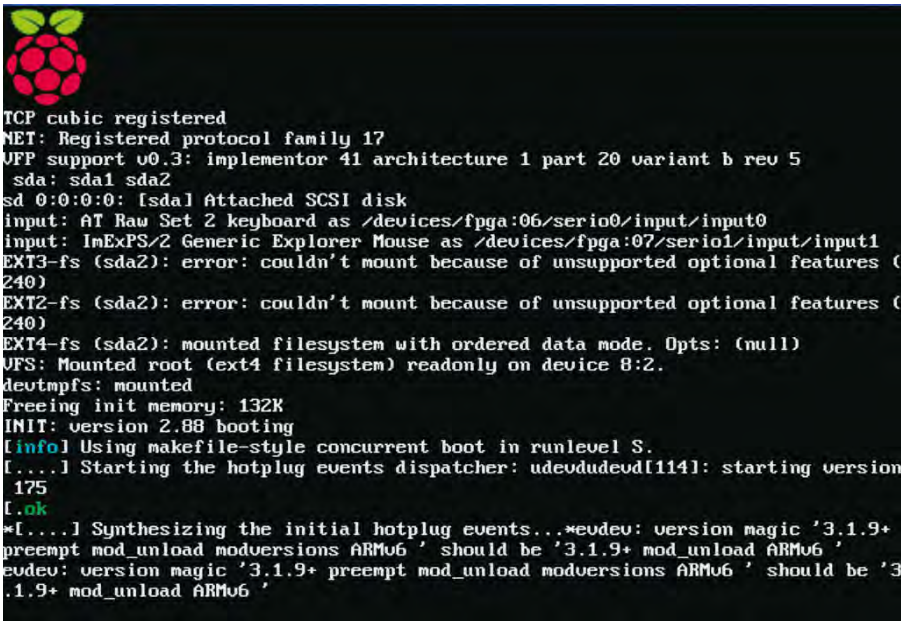
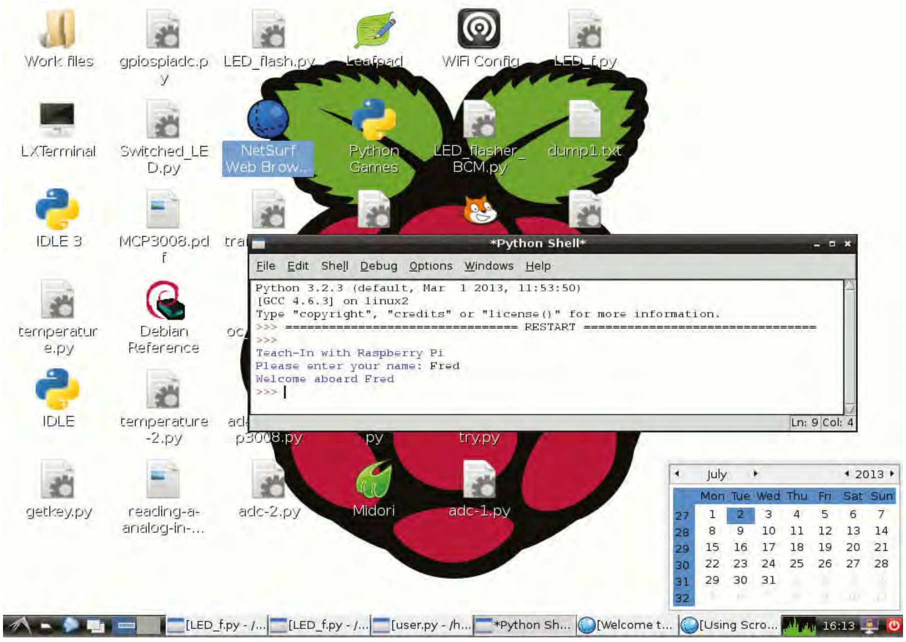


Fig.1.13. The boot screen

Shutting down your Raspberry Pi safely

Just like any other computer, it's very important to shut down your Pi properly. Failing to do this can cause all sorts of problems, such as corrupting the file system on your SD card. You can shut down the Pi from within the GUI by clicking the red power icon on the bottom right corner of your screen. Alternatively, you can execute a system shut down command from a terminal window; select 'LXTerminal' from 'Accessories' in the program menu (the start button equivalent) or click on the desktop icon and enter the following command `sudo shutdown -h now`. Following this command, the system will take a few more seconds to shut down safely.



A brief guide to config.txt

A number of the hardware settings for the Raspberry Pi are held in a plain text file named **config.txt** found in the root directory of the SD card. The file can be easily edited in Windows or OS X, allowing you to produce hardware changes from another system. This is particularly useful if you have an issue preventing booting or proper use of the Pi; for example, setting the display output (a very common problem users find when they try to boot for the first time).

```
# Set stdv mode to PAL (as used in Europe)
sdtv_mode=2
# Force the monitor to HDMI mode so that sound will be sent over HDMI cable
hdmi_drive=2
# Set monitor mode to DMT
hdmi_group=2
# Set monitor resolution to 1024x768 XGA 60Hz (HDMI_DMT_XGA_60)
hdmi_mode=16
# Make display smaller to stop text spilling off the screen
overscan_left=20
overscan_right=12
overscan_top=10
overscan_bottom=10
```

Fig.1.17. Example of config.txt, courtesy of elinux.org

You should be able to locate the **config.txt** file in the root directory of the SD card. If one does not exist, you can simply create a new plain text file in Notepad or similar and name it 'config.txt'. Alternatively, there are many example files on the Internet that you could download and amend to meet your requirements. Note that lines starting with a # are 'comments', meaning that they are descriptors to be used when editing, but not literal instructions/settings for the Pi. Note also that some of the example **config.txt** files include all of the available settings and you simply remove the # from the beginning of the setting line that you wish to activate.

There are numerous settings that can be configured from **config.txt**; we have described some of the main areas of settings below. For specific information about individual parameters and settings, visit: **http://elinux.org/RPiconfig**

Computer ▸ Secure Digital (D:)

Name	Date modified	Type	Size
bootcode.bin	08/02/2013 23:21	BIN File	18 KB
cmdline.txt	09/02/2013 00:27	Text Document	1 KB
config.txt	09/02/2013 00:27	Text Document	2 KB
fixup.dat	08/02/2013 23:21	DAT File	6 KB
fixup_cd.dat	08/02/2013 23:21	DAT File	3 KB
fixup_x.dat	08/02/2013 23:21	DAT File	9 KB
issue.txt	09/02/2013 02:42	Text Document	1 KB
kernel.img	08/02/2013 23:21	Disc Image File	2,737 KB
kernel_emergency.img	08/02/2013 23:21	Disc Image File	9,385 KB
start.elf	08/02/2013 23:21	ELF File	2,632 KB
start_cd.elf	08/02/2013 23:21	ELF File	556 KB
start_x.elf	08/02/2013 23:21	ELF File	3,547 KB

Fig.1.18. SD card files

Memory settings

These advanced settings adjust how the memory is allocated to hardware devices.

Video settings

These commonly tweaked settings are used to set the display mode and output settings, such as HDMI mode, resolution, refresh rate and over scan. If you're unsure about supported resolutions, try a basic, low-resolution setting initially and then try progressively higher settings until you achieve the optimum result. If you are using a VGA adapter, you may find that other users have posted the **config.txt** settings that worked for them on reviews or forum posts.

Licenced codecs

The hardware of the Pi is capable of decoding various audio/video codecs. However, some of these must be licenced in order to function; eg, MPEG 2 and VC-1. Licence keys are specific to each processor and can be purchased for a couple of pounds. These are particularly useful if you use the Pi as a multimedia device.

Boot settings

These settings relate to how the Pi boots; eg, boot delays, splash screen, kernel handling.

Overclocking

The clock speed of the processor may be altered, along with various other processor settings, in order to achieve faster processing. Be warned that getting these settings wrong may cause overheating and/or permanent damage to your processor. We'll look at overclocking in a later edition of *Teach-In 2014*.

Python Quickstart

Python is a powerful, but easy-to-learn general-purpose programming language. Python is freely available and will run on a wide variety of computer systems. So, if you don't yet have a Raspberry Pi, you can still install and use the software on virtually any Windows, Linux or OS-X-based computer system. Python is also well documented, and you will find a great deal of useful information on the web, as well as a number of good books, including those specifically aimed at the Raspberry Pi user. The official library and language references for the latest version of Python (3.3) are available for downloading from: **http://docs.python.org**. However, if you find these documents a little technical, take a look at the Python Tutorial first.

To keep things simple and because you might not have Python up and running yet on your Raspberry Pi, the examples that we will be describing in this first *Teach-In 2014* have been designed to run on an existing Windows, Linux or OS-X-based computer system with Python installed. The complete Python package for Windows, Linux or OS-X can be freely downloaded from the Python Software Foundation's website: **www.python.org/getit** (the PSF is a non-profit corporation that holds the intellectual property rights behind the Python programming language). In next month's *Teach-In 2014*, we will be moving on to show you how to write and test your Python code directly on the Raspberry Pi.

Python versions

The Python programming language was originally conceived in the late 1980s, and its implementation was started in December 1989 by its principal author, Guido van Rossum. Python 1.0 appeared in January 1994, and versions 2.0 and 3.0 were released in 2000 and 2008. The current version of Python is 3.3. This was released on 29 September 2012. It is important to note that Python 3.0 is not backwards-compatible with earlier versions and, while version 2.7 (July, 2010) is still in common use, we recommend that you use the latest version for your own programs. Note that all of the code in our *Teach-In 2014* series has been written in Python 3.3, and may not run with earlier versions.

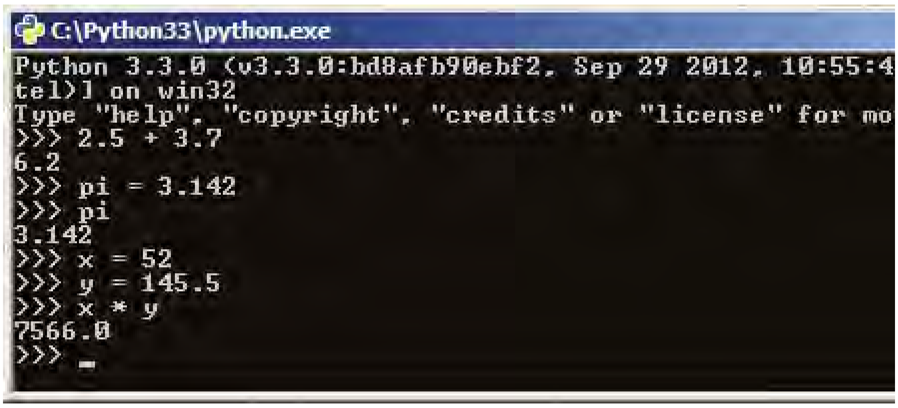


Fig.1.20. Using the Python shell in 'calculator mode'

Interpreted versus compiled languages

It is important to realise that Python is an interpreted language. This means that it will execute your code directly without you having to compile the code into a file that is separately executable. Interpreted languages are easy to use and platform independent. Compiled programs, on the other hand, need to be compiled into code that can be understood on whatever computer system they are run.

The advantage of converting the entire source code into machine executable code in one operation is that a program will operate very much faster than it would on a statement-by-statement basis. Examples of compiled languages include C++ and Pascal. Finally, you need to be aware that when you install Python you are actually installing a set of standard libraries that your code can make use of – *as well as* the Python interpreter itself.

Statements

As with all interpreted languages, Python code is executed one statement at a time. Furthermore, a Python statement can be executed directly by simply typing it within the command line environment (known as the 'Python shell' – see Fig.1.20). This is sometimes referred to as 'calculator mode' because you can simply type an arithmetical expression and have it evaluated when the enter key is depressed. It is also possible to assign values to variables and use these within your expressions.

Using the Python shell in 'calculator mode'

Python can be used as a simple calculator by typing an arithmetical expression directly after the shell prompt, >>>. For example:

```
>>> 2.5 + 3.7
```

When you press the return or enter key, you should be rewarded with the result, 6.2 (as shown in Fig.1.20). You might like to try a few more calculations using the +, –, * and / keys. Note that you can use brackets to ensure that more complicated calculations are carried out in the right order. For example, the following will produce different results:

```
>>> 2 + 3 / 5
2.6

>>> (2 + 3) / 5
1.0
```

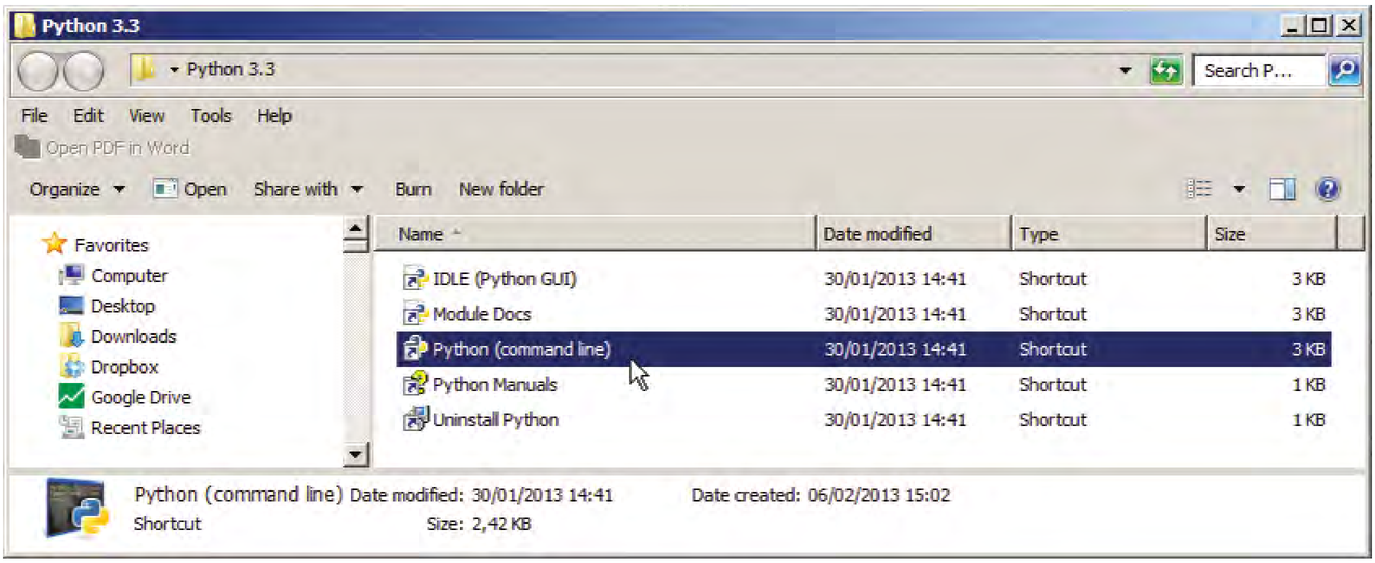


Fig.1.19. Selecting the Python command line shell from Windows

Assigning a value to a variable

In order to give a value to a variable, you simply need to use a statement of the form `variable = value`. For example, `pi = 3.142`. We can also assign a value to a variable by using an expression, such as: `pi = 22/7`. In this case, the expression on the right-hand side of the equation is evaluated and then the result is given as a value to the variable. The expression can be quite complicated and can involve all of the allowed operators, such as: + (addition), – (subtraction), / (division), and * (multiplication). To print the current value of the variable from the Python shell, you only need to type the name of the variable and then press the enter key.

In 'calculator mode', you can enter the values of as many variables as you need. For example, `x = 5` and `y = 145.5` and then evaluate expression, such as `x*y`, as shown in Fig.1.20.

Printing the value of a variable

To print the value of a variable, you can use the `print` statement. For example, `print x`. Here's a simple example of assigning an expression to a variable and then printing the result:

```
x = 3.142
print x
```

Before moving on, it's well worth taking a little time out to experiment with the Python shell operating in 'calculator mode'.

Creating Python programs with IDLE

The Python shell can be used for simple tasks, but there's a need to have some means of entering and saving a series of statements so that you don't need to type them every time you use them. A Python program or 'script' takes the form of a series of Python statements that are stored in a file. In order to enter, save and recall the file we need the services of an editor. An added bonus would be the ability to execute the file and test it without having to exit to the Python shell. We can do all of this with the aid of a utility such as 'IDLE'.

IDLE stands for 'Integrated DeveLopment Environment'. The program is packaged with most Linux distributions and it is completely written in Python, together with an additional toolkit that provides the graphical user interface (GUI). IDLE comprises:

- Multi-window text editor that provides syntax highlighting, auto-completion and smart indentation
- Python shell with syntax highlighting
- Integrated debugger with stepping, breakpoints, and several other useful features.

To start IDLE, you simply need to locate it in your Python folder and then click on it. Having done that, let's enter, test and save a simple program that will convert an ordinary number into its binary and hexadecimal equivalent. To start a new program you can click on the File menu and then select New from the drop-down menu.

Next, you will need to enter the following program statements (note that in the IDLE editor you no longer have the >>> prompt before each line of code):

```
# A simple example
xString = input("Enter a number: ")
x = int(xString)
print(bin(x))
print(hex(x))
```

If you have any previous programming experience, the code isn't too difficult to understand, but for the benefit of anyone that is completely new to programming, we shall briefly explain each line of code.

- The first line of our program starts with a # character and is simply a comment. When the time comes to execute the program everything after the # is ignored. Liberally commenting your code is good practice and always a good idea – particularly if you ever need to modify your program or if you want to pass it on to someone else.
- The second line of the program assigns an input value (in this case an alphanumeric string from the keyboard) to the variable `xString`)
- The third line of the program assigns a value (in this case an integer number) that is the numerical equivalent to the `xString` variable
- The first `print` statement prints the numerical value in binary format, while the second `print` statement prints the same numerical value in hexadecimal form. We thus have a simple program that will let you find the binary and hexadecimal equivalents of a number entered from the keyboard.

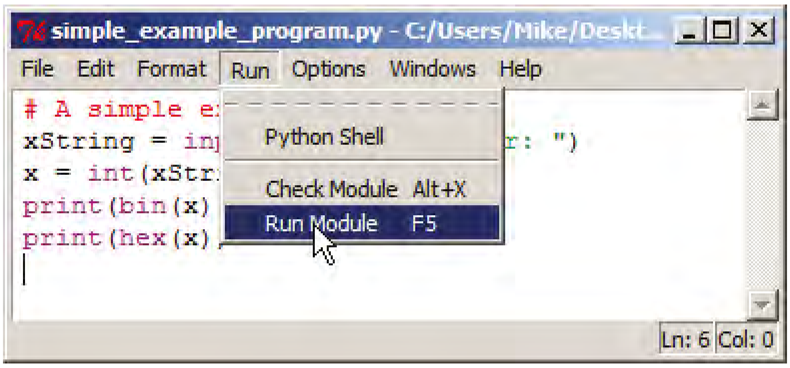


Fig.1.22. Just click on Run and then Run Module in order to run the program

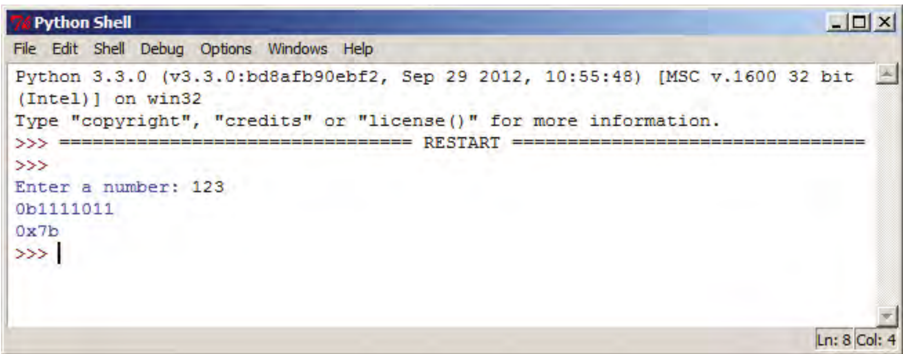


Fig.1.23. The results of executing the example program appear in a conventional Window rather than in a more basic command shell like that shown in Fig.1.20

In the example that we've just described (and when using Python 3.3, or later) it's important to note the use of the brackets that enclose the function and variable that we are passing into the `print` function. So, in the case of the first `print` statement, the result of

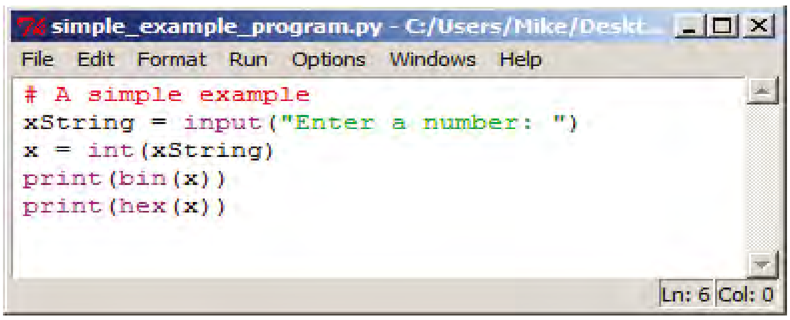


Fig.1.21. Our simple example Python program ready to run

applying the bin() function is passed into the print() function and the result is then displayed as text on the screen. This convention is something that you need to remember. Your entered program should look like the one shown in Fig.1.21. Finally, don't forget to save your finished program before you run (see Fig.1.22)

Programs, modules and libraries

A Python program (or module) is simply a series of Python statements that can be entered using a plain text editor (or using the editor supplied with IDLE) and then saved (usually with a .py file extension). Python modules provide you with a way of grouping a number of Python functions together so that they can be imported into your program code and easily reused. Some useful Python modules are shipped as part of a standard Python installation, but others can be downloaded from the web. Alternatively, you can easily write and test your own modules and then add them to your library of reusable code.

The Python standard library includes a number of useful modules. Some of the handiest are:

- datetime – for handling and formatting date and time information
- math – mathematical functions (such as max, min, sine, cosine and tangent)
- random – for generating random numbers in a given range
- string – utilities for handling alphanumeric character strings (eg, 'Hello world')

Within each of these modules is a number of predefined functions. So, for example, within the datetime module you will find the today function, which will tell you what the date is. The following code fragment will give you an idea of how this is used (note that we have to import the datetime module before we can make use of it):

```
import datetime
today = datetime.date.today()
print('Today is:', today)
print('Year =', today.year)
print('Month of the year =',
today.month)
print('Day of the week =',
today.day)
```

If you enter and then run the code you will be rewarded with the current date (in year, month, and day format) followed by the year, day and month. Why not give it a try?

Next month's slice of Pi

Next month, *Teach-In 2014* is all about connecting the Raspberry Pi to the real world. Our *Pi Class* will take you on a comprehensive tour of the GUI and the applications that are bundled with the Raspberry Pi's operating system. *Pi Project* will explain how the Raspberry Pi's general purpose input/output (GPIO) interface works and provides you with four simple I/O projects, together with the necessary Python code to make them work. Our feature for programmers, *Python Quickstart*, will provide you with an introduction to comparisons and loops. Last, but by no means least, *Home Baking* will explain how you can connect your Raspberry Pi to the Internet.

Ingenuity Unlimited

We pay between £10 and £50 for all material published, depending on length and technical merit. We're looking for novel applications and circuit designs, not simply mechanical, electrical or software ideas. Ideas must be the reader's own work and **must not have been published or submitted for publication elsewhere**. The circuits shown have NOT been proven by us. **Ingenuity Unlimited** is open to ALL abilities, but items for consideration in this column should be typed or word-processed, with a brief circuit description (between 100 and 1000 words maximum) and include a full circuit diagram showing all component values. **Please draw all circuit schematics as clearly as possible.** Send your circuit ideas to: Ingenuity Unlimited, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU.

Email: editorial@epemag.wimborne.co.uk

Your ideas could earn you some cash and a prize!

Our regular round-up of readers' own circuits

WIN A PICO USB DrDAQ DATA LOGGER PH KIT WORTH £139

• Use DrDAQ as a data logger • Use DrDAQ as an oscilloscope • Use DrDAQ as a signal generator • Built-in sensors for light, sound and temperature • Measure pH – just plug in any standard pH electrode • Sockets for external sensors • Digital outputs to control external devices • USB connected and powered • Use up to 20 USB DrDAQs on a single PC .

If you have a novel circuit idea which would be of use to other readers, then a Pico Technology USB DrDAQ Data Logger PH Kit could be yours. After every 20 published IU circuits, the best entry will be awarded a USB DrDAQ Data Logger PH Kit worth £139. In addition, a runner up will be awarded with a USB Dr DAQ Data Logger woth £99.

Simple chess clock – your move!

It is often a challenge to replace a mechanical device with an electronic version, as simple mechanical designs can become cumbersome or expensive when realised in an electronic version – consider the case of a corkscrew!

An intriguing device is the chess clock, which is two simple mechanical clocks in a single case; with a player-controlled changeover mechanism such that only one clock can run at a time. Players make a move when their clock is running, then stop their clock by starting the opponent's. Each clock has a flag, which is raised by the minute hand at five minutes before the hour, and then drops on the hour. Chess rules require a player to make a defined number of moves within an hour of playing time ie, before the flag drops. There is also a version of chess where each player's flag is raised (by setting the clock hands appropriately), and the player whose flag drops first (ie, after five minutes of play) is the loser, unless the game has already ended in mate.

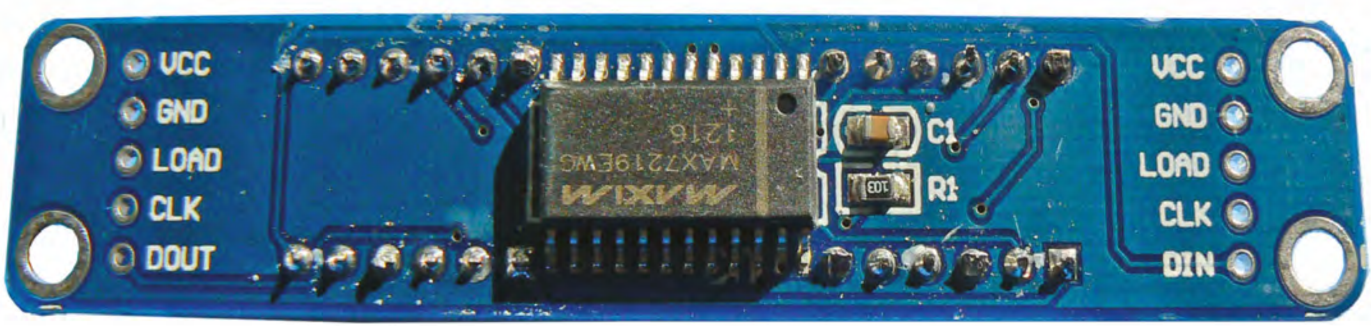
Implementing such a device electronically and yet simply was extremely difficult before the advent of the microprocessor, and particularly before the integrated microcomputer. The author spent many hours designing a chess clock using a calculator chip with gates to produce key presses, but a meaningful display was always a problem, and the support logic became unwieldy.

Not surprisingly, there have been many electronic versions, but these have been remarkably complex or tedious to construct. What has really made this electronic chess clock outstandingly simple is, surprisingly, the availability of a serial-driven LED display at a reasonable cost.

Serial-driven display

The display is two four-digit LED displays mounted together on a small PCB, with a Maxim MAX7219 driver on the rear of the PCB. A three-wire interface is required, together with +5V and ground. The major advantage of such a device is that everything – data retention, multiplexing, display brightness control, decoding and blanking – is handled by the driver – thus the processor only sends data (asynchronously) when necessary, and is otherwise freed from routine housekeeping such as refreshing the display.

Fig.1. Rear view of the serial-driven 8-digit display



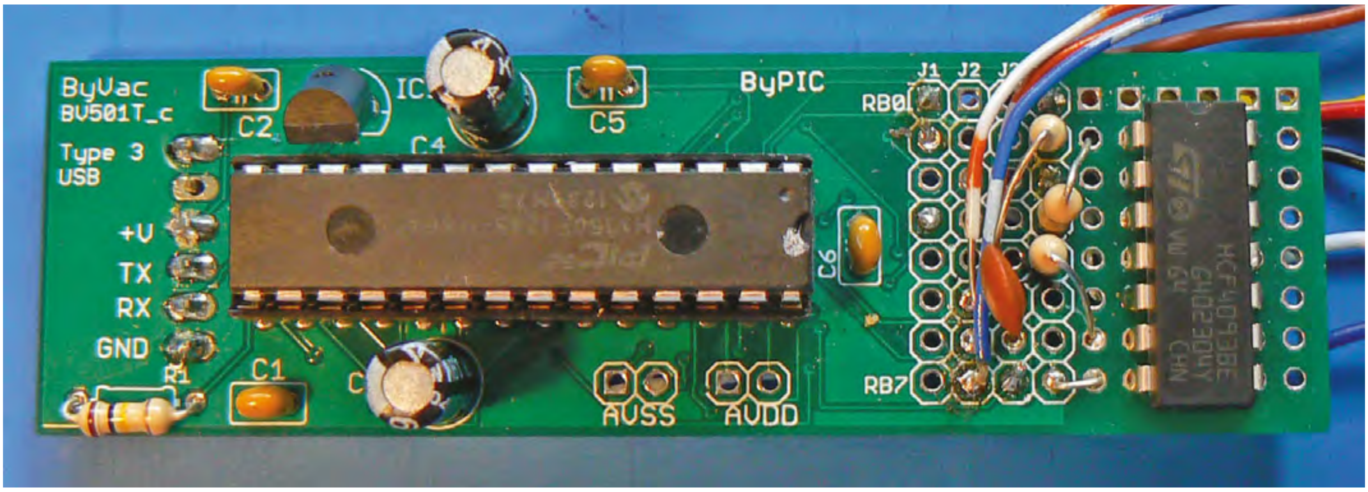


Fig.2. BV501T board with CD4093

The full capability of the MAX7219 driver is best understood by reading its 17-page specification, but in summary, each digit of the display is addressed separately; the display data may be a number which is decoded into a seven-segment digit, or a direct addressing of the seven segments; each decimal point may be controlled separately; any digit position may be blanked. The drive interface is compatible with SPI (Serial Peripheral Interface bus), but requires an additional LOAD line, and the MAX7219 accepts 3.3V logic signals while itself running on 5V. One caution – the data out signal from the display (which is not used in this application) is at full +5V level.

These integrated displays are readily available from international suppliers on eBay at a total price of around £4 each, but the supplier keeps changing – it may be that each supplier buys a batch and sells that off, and then has to wait for another batch to become available.

Other components

Many popular microcomputers can be used in this application, since there is a total of only seven signal lines – four player-controlled inputs and three outputs to drive the display. My project is described with a particular device, but it is an easy matter to adapt the principle to use any suitable microcomputer.

The processor used here is a BV500, because it provides an inbuilt function-based language that is self-compiling and is similar to BASIC in its statement syntax. As a consequence, the program is extremely easy to understand and to modify if desired. The inbuilt oscillator provides an accurate timebase without external components. The processor can be obtained with a small PCB, which contains a voltage regulator to provide its 3.3V supply, together with all other components to make the processor functional. There is also a small prototyping area which can hold the single logic package required to complete the electronic components.

There are significant advantages in using an external RS flip-flop, here constructed from logic gates, to retain the state of the player changeover switch. This approach removes the need to detect the transition in real time, so simplifying the processor logic, and the outputs are available to drive other indicators if desired. The spare gates may be used to buffer the outputs if necessary; since their inputs must be connected, this is an effective solution.

There is a diode connection between the reset and the 'White to play' buttons; this ensures that White's clock is running after a reset, as White always starts a game.

The clock is powered by a wall-mounted 5V supply; there is thus no on/off switch.

Operation

All information is presented on the 8-digit display, which is treated as two blocks of four digits. The leftmost block is White's clock (so-called for convenience – the actual ownership depends on where the clock is placed); the description applies equally to the other (Black's) block. The leftmost digit has its segments directly driven, with seven-segment decoding applied to the other three digits. The decimal point on the third digit (from right) is permanently on, so the display normally shows (blank) X.YY, representing either H.MM when clock is counting up (normal mode) or M.SS when playing in 'five minute total' countdown mode. When the player's clock is running, the leftmost decimal point flashes on alternate seconds. In the normal mode the player's time is incremented every second, although the display only shows the elapsed time in hours and minutes. The leftmost digit represents the flag, and shows 'F' for 10 seconds at the end of

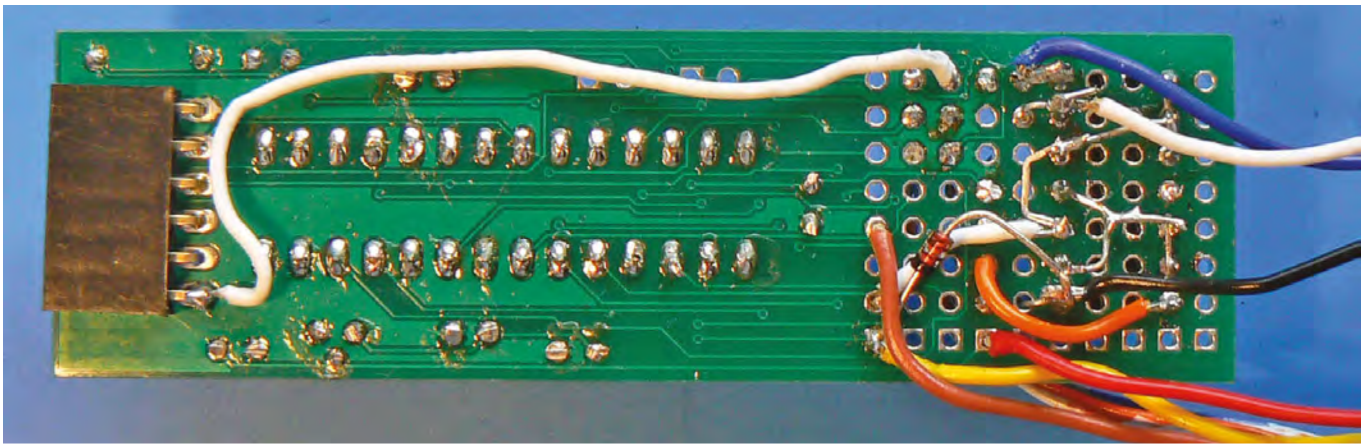


Fig.3. BV501T board from rear showing gate interconnections and all external wiring. The white wire running across the board is a DTR connection, and is used during programming

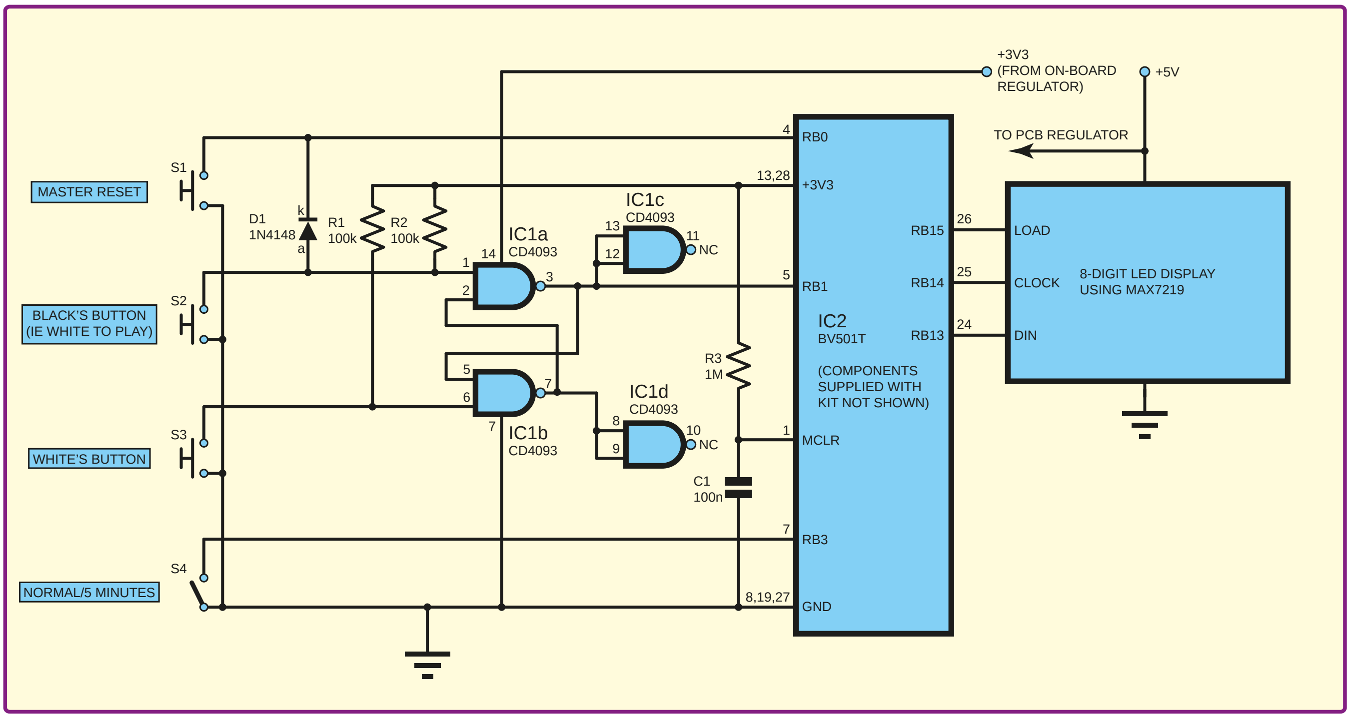


Fig.4. Chess clock circuit diagram

every hour in normal mode, or shows 'F' and locks the count at zero when time expires in countdown mode.

Reset initialises the displays either to 0.00 0.00 or to 5.00 5.00, depending on the mode selected, and starts White's clock running immediately.

Software

The software file (7219_ch_clk1.bas) is available for download from the October section of the *EPE* website. It uses the ByPic library functions 'Rookie' to simplify access to the processor registers, and to handle the SPI interface to the display. This is fully documented, with tutorials, on the ByPic site (**www.bypic.co.uk**). Consequently, all processor operations are calls to the appropriate functions, and no knowledge of the register addresses is required.

The clock program is believed to be fully commented and self-explanatory. However, the method of integrating the program into flash memory is not described, because it is specific to this microcomputer; it is readily available from the ByPic documentation.

The method of selecting the mode of operation by use of the case statement makes it very easy to add other modes of time control; only the logic for choosing the mode needs to be altered to reflect the method of choice, as suggested under further developments.

Construction

Construction is very much a matter of individual taste, and can be as complex or simple as desired. With the emphasis here on simplicity, the clock is mounted in a small ABS box, 100 × 75 × 40mm. The display is fitted into a 60 × 14 mm slot cut in the front face. The player changeover switches are simple pushbuttons; the actuators are buttons (the type intended to be covered with fabric) with a foam inlay to provide height, glued on top. The reset and mode switches are on the side, together with the power connector, to ensure that reset is not actuated accidentally.

Stray pickup has been noted to affect the start-up sequence of the clock; and for this reason the MRST (master reset) pin has a long time constant RC network to allow the MAX7219 to start before the processor begins its program. Depending on the construction and the power supply used, it may be necessary to ensure that the negative side of the supply is grounded. (A metal case would probably help.)

Fig.5. Preparing the actuators

Fig.6. Attaching the actuators to the buttons

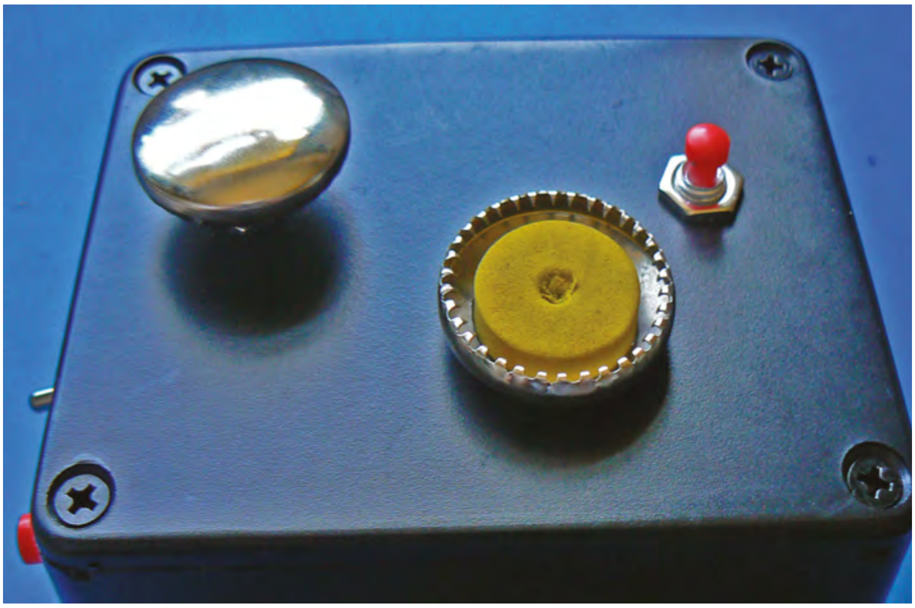


Fig.7. Clock in normal mode – White's flag has dropped after one hour; Black has used eight minutes more

Simple chess clock components list

BV501T board KIT2 (ByVac Electronics)
8-digit LED display with integrated MAX7219 driver (eBay suppliers)
CD4093 (or CD4011)
IN4148 or similar

Resistors
100kΩ 0.1W (2 off); 1MΩ 0.1W

Capacitors
100nF ceramic
Plug-in power supply: 5V 0.25A
Socket 2.5mm power, or as required to fit power supply
Push button (3 off)
Actuator (see text) (2 off)
Switch SPST
Case

Further developments

The system described was designed to be as simple as possible. If this principle is abandoned, then there are many possible enhancements, of which a few are outlined here.

Several other time control schemes are used; for example, in chess leagues. It is a simple matter to code and add any of these if desired. The method of selection becomes interesting – is a particular time scheme to be programmable, or should it be chosen from a pre-programmed menu; eg, step through a series of modes? The BV500 microcomputer has plenty of spare I/O lines, so the choice becomes one of efficiency and ease of use. The time control scheme need not even be hard-coded; this particular microcomputer allows easy storage of parameters in Flash memory under program control. Hence, at the expense of program complexity, other time control modes could be programmed and stored.

The LED display was selected for clarity, and because the necessary indications could be provided without extra components. The flashing decimal point, as a means of indicating which player's clock is running, was chosen to be both clear and not distracting. The spare logic gates could be used to drive additional LEDs for a more visible indication, although these would not flash without additional circuitry (or perhaps using an extra line from the processor?). If more information is required on the display, readily available low-cost LCD displays can provide two lines of 16 characters or four lines of 20 characters; either can be controlled using only six lines and 3.3V logic levels. However, the size of the LCD characters may be a drawback; a possible solution is to use the LED display for the times, and the LCD for programming or mode display.

Naturally, the display can be even more complex – a 320 × 240 pixel screen offers many possibilities, but this is a long way from the original simple scheme.

The mechanical design also offers many permutations. The switches could be capacitive or touch-sensitive, although it is likely that chess players prefer a positive movement –observation shows that many players actuate the clock without looking at it, so some tactile feedback is desirable. One possibility, which mimics the mechanical clock closely, is an internal pivoting bar like a seesaw, with long travel actuators; this could use a magnet and reed switch/sensor arrangement, and has the additional advantage that the actuators are clearly *up* or *down*.

Ross Henderson

READOUT

Matt Pulzer addresses some of the general points readers have raised. Have you anything interesting to say? Drop us a line!

WIN AN ATLAS LCR ANALYSER WORTH £79

An Atlas LCR Passive Component Analyser, kindly donated by Peak Electronic Design Ltd, will be awarded to the author of the Letter Of The Month. The Atlas LCR automatically measures inductance from 1mH to 10H, capacitance from 1pF to 10,000μF and resistance from 1Ω to 2MΩ with a basic accuracy of 1%. www.peakelec.co.uk

All letters quoted here have previously been replied to directly

Email: editorial@wimborne.co.uk

★ LETTER OF THE MONTH ★

More MEMS ingenuity

Dear editor
Like Mr Naylor (*Readout*, *EPE* June 2013), I too had issues with soldering the MEMS chip for the *Digital Spirit Level*, resulting in a blank display when powered up. Several attempts at re-soldering failed to correct this. Reluctant to abandon the project, I removed the chip by carefully cutting through the PCB tracks round the chip (to avoid lifting any track other than under the chip), then easing the chip off the board with a craft knife blade. This revealed a couple of pads that had failed to solder. After a bit of cleaning up, to remove the bits of pad that had soldered and had lifted with the chip, I soldered wires directly to the pads of the IC, using 40swg enamelled copper (anything thicker pulls off the chip pads when you try to bend it). A good illuminated magnifier is useful for this exercise.

The resulting 'spider' was connected to the PCB by scraping the lacquer off suitable areas of the PCB tracks, positioning the MEMS chip and soldering the wires to the tracks. The spirit level then worked, so the chip and its wires were covered with some silicone sealant to reduce the risk of damage to the fine wires, and to make sure the chip didn't move.

My experience demonstrates, among other things, that the MEMS chip is very resilient – it had undergone a lot of soldering attempts, and yet still works normally.

There is another possible use for the spirit level, as an altitude indicator for an astronomical telescope (when using an alt/az mount). Ideally, the level would have a manual power switch, rather than automatic shut down, and a dimmable display. Both should be easily obtained when I have time to look at the code (dimming probably by altering the pulse width to the FETs). The red LED display is ideal for astronomy use, red having less effect on 'night vision' than other colours.

Perhaps a future *EPE* project could be an azimuth indicator for a telescope? Better still, combine the two for an RA/declination indication.

Love the PIC projects – keep them coming!

Thomas Thorley, by email

Matt Pulzer replies:

Well done Thomas, that's a great solution to the MEMS soldering problem. I like the telescope mount application – do let us know if you build one!

Relay reliability

Dear editor
Is there any magic formula that can be used to match up a plain old relay with a relay socket?

I've built a device for a relative that controls an external mains bell. I switch the bell using a mains relay: 12V to the coil switches mains power to the bell. The device has worked fine for a few years, but then the relay broke. I originally soldered the relay to a PCB, so I had all the inconvenience of decoupling the unit, taking it to the bench then desoldering the old relay and soldering in a new one.

Sadly, this was a few years ago and the new one has just bitten the dust. I don't want any more hassle, so I'd like to buy a relay plus socket to make future maintenance a doddle.

I'm confused by the options out there; any help appreciated when you get a spare moment please. Are there 'better' sockets that are more 'future proof'?

Mungo Henning, by email

Matt Pulzer replies:

Welcome to the frustrating world of device physical compatibility! I wish I could help, but my relay socket knowledge is vanishingly small. You might do well to contact a large, well-established firm like Farnell. Are there any relay experts out there who could help Mungo?

Tracing Magenta Electronics files

Dear editor
I've been looking on your website for the files ib1.asm, ib2.asm and ib3.asm, which were provided on the *EPE IceBreaker* disk in 2000.

These are test files for the development board I bought from Magenta to drive an LED, stepper motor and LCD display using a PIC16F877.

I contacted Magenta to hopefully receive a copy of these files, but heard nothing back.

I checked the contents of ICEbreaker/Ice10.zip on **ftp://ftp.epemag.wimborne.co.uk/pub**, but found only the ICEBreaker software itself, not the example source code.

I am not looking for source code for the ICEBreaker, just source code for the examples ib1.asm, ib2.asm and ib3.asm.

I am trying to resurrect this project to help start a business, but cannot get much out of it without the source code. Do you have a copy of these files?

Rich Webster, by email

Alan Winstanley replies:

Hello Rich,
This is a very old commercial project originally produced by Magenta Electronics. I checked the archives and the only software we ever got was the icebreaker.exe and Help files in ice10.zip, which are hosted on the old EPE download page at epemag.net. Unfortunately the ib.asm test routines were not included in the download, as you have found, and would have been supplied direct by Magenta who owned the design.*

The project states that the asm files were included on (floppy) disk (presumably supplied by Magenta). Unfortunately, despite what Shop Talk states (page 202, March 2000), I can advise that the demo test files weren't ever made available by FTP.

Magenta were usually very helpful, but they have diversified into their own products and have very little involvement in hobby electronics

these days. The only possibility is that if you still have a floppy disk, find someone with a PC or laptop FDD and have it read that way. A USB floppy drive can be obtained on eBay, or just get an IDE floppy drive if you have IDE. If the disk is mission-critical to your new business then I guess the cost will be trivial if it means that the Icebreaker can be resurrected.

It is also possible that other EPE *readers could help. Many of them have a long memory, so why not try our forum at* **www.chatzones.co.uk** *and post into the* EPE Chat Zone. *As a last resort, I will offer to try reading your floppy disk on my own hardware.*

I hope the above is of interest.

New physics?

Dear editor
I've been busy with unconventional radiation detectors and have discovered what I believe is a completely novel effect undocumented in the scientific literature.

Back in early 2008, I wrote a message on the 4HV forum (**http://4hv.org**) asking if anyone else had noticed the effect of some cheap CMOS cameras being very sensitive to alpha particles.

The original effect was noticed with a B&Q monochrome camera with its case removed so that the chip was exposed. I added a mica window painted with silver paint to light proof the assembly. This later evolved into the use of pyrolytic graphite (PG) as a window material, which seemed at the time to pass more alphas from Am-241 (americium) and Th-232 (thorium) than the mica, comparable with a bare sensor.

The thinned down and cleaned graphite worked best here, and replacing the stock 13.5MHz crystal with a 4MHz one from a broken optical mouse worked very well; I could see individual particle tracks on my 'scope due to the lower clock speed allowing more time for charges to accumulate in the sensor elements, showing a veritable pincushion of flashes. I also noticed this with an SBM21 Russian Geiger-Muller tube, which is thin-walled. Although the SBM21 was originally designed to pass beta and gamma, it will sometimes show alpha sensitivity at slightly elevated voltages. The effect of PG being alpha transparent is interesting, as it should block nearly every particle, my experiments showed 70% and higher made it through.

My working hypothesis at this time is alpha entanglement, ie adjacent particles could be being affected by the diamagnetic field at the impact surface, which allows them to get through without being absorbed as well. There is a reference to the use of PG as an X-ray spectrometer element, but nothing about alpha particles that I am aware of. If so, then this constitutes the beginnings of a 'quantum sensor' and thus new physics. I am currently building an improved version of this sensor using an Aixiz module with a PG window, as this is light tight as well as electrically shielded. A bare piece of silicon solar cell with a high reverse bias (ie, 30V) would avoid the use of an expensive camera module for dosimetry and draw much less power in operation.

As an aside, PG also offers damage protection for conventional alpha-end window tubes.

Andre J de Guerin, (aka 'Conundrum' on 4HV.org and 'BotherSaidPooh' on hackaday.com, by email

Matt Pulzer replies:

Thank you Andre, a most interesting letter. I must confess I am not qualified to offer an opinion on your 'new physics' – perhaps a knowledgeable reader can comment?

Thank you!

Dear editor
Folks, Roger in Iowa (in the US) here: just wanted to drop you a line to let you know I found your article on reclaiming CD/DVD motors and rebuilding them into high-power brushless motors to be absolutely superb.

A friend showed me the article, since sadly I don't receive the magazine. It really is a top-shelf piece of up-cycling design, and a very descriptive and entertaining article, too.

It has inspired me to check on purchasing your publication here in the states.

Last, but not least, those resistance and capacitance decade boxes also look like intriguing projects. Finally, a magazine that doesn't feel the need to 'dumb things down' for electronics enthusiasts!

Thank you for what you do! And well done, friends!

Roger Benz, Dubuque, Iowa, USA

Matt Pulzer replies:
Thank you for your kind words and I'm delighted to hear that EPE has trans-Atlantic fans.

You may be interested to know that you can subscribe online for not a lot of $!
http://pocketmags.com/viewmagazine.aspx?titleid=1466&title=Everyday+Practical+Electronics

IF YOU HAVE A SUBJECT YOU WISH TO DISCUSS IN READOUT PLEASE EMAIL US AT:

editorial@wimborne.co.uk

EVERYDAY PRACTICAL ELECTRONICS

UK readers you can SAVE 94p on every issue of *EPE*

How would you like to pay £3.46 instead of £4.40 for your copy of *EPE*?

PRICE FREEZE!! BUY NOW!

Well you can – just take out a one year subscription and save 94p an issue, or £11.28 over the year. You can even save £1.15 an issue if you subscribe for two years – a total saving of £27.60.

Overseas rates also represent exceptional value.

You also:

- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Ensure your copy, even if the newsagents sell out**

Order by phone or fax with a credit card or by post with a cheque or postal order, or buy online from www.epemag.com (go to the Online Shop).

EVERYDAY PRACTICAL ELECTRONICS

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in the UK: 6 months £21.95, 12 months £41.50, two years £78.00; Overseas: 6 months £25.00 standard air service or £35.00 express airmail, 12 months £48.00 standard air service or £68.00 express airmail, 24 months £91.00 standard air service or £131.00 express airmail.

Cheques or bank drafts (in **£ sterling only**) payable to *Everyday Practical Electronics* and sent to EPE Subs. Dept., Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU. Tel: 01202 880299. Fax: 01202 843233. **Email:** subs@wimborne.co.uk. Also via the **Web** at: **www.epemag.com**.

Subscriptions start with the next available issue. We accept MasterCard, Maestro or Visa. (For past issues see the Back Issues page.)

ONLINE SUBSCRIPTIONS

Online subscriptions, for reading the magazine via the Internet, £19.99 for one year, visit **www.epemag.com** for more details.

SUBSCRIPTION ORDER FORM

☐ 6 Months: UK £21.95, Overseas £25.00 (standard air service), £35.00 (express airmail)
☐ 1 Year: UK £41.50, Overseas £48.00 (standard air service) £68.00 (express airmail)
☐ 2 Years: UK £78.00, Overseas £91.00 (standard air service) £131.00 (express airmail)

To: Everyday Practical Electronics, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel: 01202 880299 Fax: 01202 843233
E-mail: subs@epemag.wimborne.co.uk

I enclose payment of £ .... (cheque/PO in £ sterling only), payable to *Everyday Practical Electronics*

☐ Please charge my Visa/Mastercard/Maestro

My card number is: ....
Please print clearly, and check that you have the number correct

Signature ....

Card Security Code .... Valid From Date.......................
(The last 3 digits on or just under the signature strip)

Card Ex. Date ....Maestro Issue No. ....

Name ....

Address ....

Post code .... Tel. ....

Subscriptions can only start with the next available issue.

INTERFACE

By Robert Penfold

Pi Eye

THE Raspberry Pi computer was introduced in the previous Interface article. This is essentially a tiny PC motherboard based on a mobile telephone chip. With the addition of a standard PC mouse and keyboard, a monitor or television, a 5V mains adapter, and an SD card containing the operating system, you have a complete PC that runs under what is usually a version of Linux. The idea behind this project was to provide educational establishments and individual users with a cheap computer that could be used to learn programming and computer interfacing.

In this column it is obviously the interfacing aspect that makes this little computer especially interesting. It has something akin to the user ports of computers of the past, such as the BBC Model B and the Commodore 64. It is called the GPIO (general-purpose input/output) port, but with its 3.3V and 5V supply lines, plus numerous input/output lines, it can be regarded as a modern equivalent of the user ports of old.

All seeing

Before moving on to a detailed consideration of the GPIO port it is perhaps worth mentioning two ports that were omitted from last month's introduction to the Raspberry Pi. These are both tiny connectors that accept miniature 15-way ribbon cables (Fig.1). The one near the SD card slot is the DSI (display serial interface) and is intended for use with an LCD screen, but support for this seems to be a bit limited at present. The other port is the CSI (camera serial interface) type, and it is for use with an add-on miniature camera board (Fig.2) that can handle up to five-megapixel still images, in addition to a few video formats.

Like the electronics in the Raspberry Pi board itself, I think that it is fair to assume that the camera is based on technology that was produced primarily for use in mobile telephones. Its performance is impressive for such a diminutive and fairly inexpensive add-on. In order to use the camera it is necessary to have an up-to-date version of the Raspian operating system installed, and it must be enabled via the configuration utility. An update is required if the configuration program does not include an option for activating the camera interface.

The camera is a popular add-on that sold out almost immediately when it was first introduced, and it certainly offers some interesting possibilities. There is no software module available to provide direct control of the camera using a programming language such as Python, but there is a free command line utility that enables video clips or still pictures to be taken. This utility also gives some control over things such as white balance and contrast.

It can be called from within a Python programme, so the camera can actually be used under software control, and it can therefore be triggered automatically via suitable hardware on the GPIO port. The files generated by the camera can be loaded into a program, and it would presumably be possible to implement things such as movement detection and simple pattern recognition.

Line by line

Using the add-on camera is something that will be considered in more detail in a future *Interface* article, and here we will move on to a more detailed examination of the GPIO port. Although it is in many ways like a conventional user port, the way in which the input/output lines are implemented is actually very different to the normal way things are handled. Conventionally, there would be something like an 8-bit bidirectional bus plus some handshake lines. These would be provided by two separate ports at different addresses in the input/output map.

Reading a single input line would therefore be achieved by reading a byte from the appropriate address and using a bitwise AND operation to mask the unwanted bits. The same basic method can be used to read two lines, three lines, etc. When writing to one line, or a small group of them, it is necessary to write a byte of data to the whole port, making sure that the states of any other output lines are not inadvertently altered in the process.

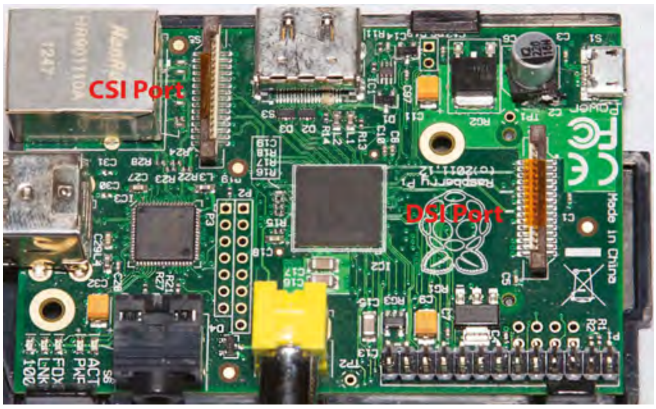


Fig.1. The CSI port (between the USB and HDMI connectors) is for an add-on camera board, and the DSI port (to the left of the SD card slot) is for use with a liquid crystal display. Both take a 15-way miniature ribbon cable

Fig.2. The camera board connected to the CSI port. The maximum resolution for stills is five megapixels, and various video formats can also be accommodated

I do not know if the input/output lines of the chip at the heart of the Raspberry Pi are organised as bytes or words, but they do not seem to be handled in this fashion by the software module that supports the GPIO port. They are presented as a number of separate lines that can be individually set as inputs and outputs. Read and write operations are handled on a line-by-line basis, which is very convenient when dealing with something like a few control outputs and status inputs. Interfacing to (say) an analogue-to-digital or digital-to-analogue converter that has some form of synchronous serial interface will require the usual mathematics, but it should be straightforward in other respects.

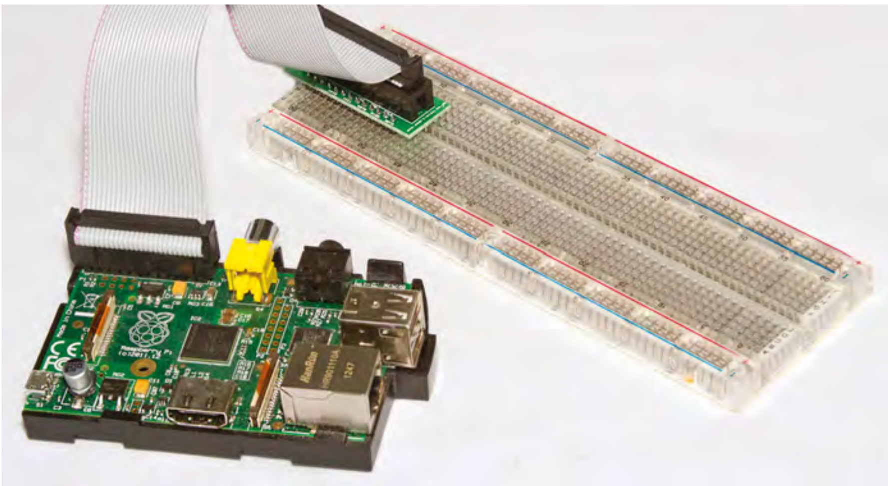


Fig.3. In order to experiment with the GPIO port safely a breadboard and connecting cable are required. This is a ready-made cable, but it should not be too difficult to make your own

Bit-by-bit read/write

Matters are less convenient when reading or writing bytes or words of parallel data. With a read operation this becomes a matter of reading the eight or sixteen bits individually and then processing this data to produce the corresponding byte or word value. With a write operation it is a matter of converting the value into binary form, and then writing the binary pattern to the output lines bit-by-bit. In other words, parallel data transfers require more or less the same mathematical operations that are needed for synchronous serial transfers. This makes parallel data transfers a bit longwinded, but not difficult to implement.

Making connections

Experimenting with the GPIO port is very easy if it is connected to a solderless breadboard. Ready-made equipment for this purpose (Fig.3) is available from specialist Raspberry Pi suppliers, but it should not be too difficult to make your own. The Python programming language and the add-on software module for the GPIO port must be installed on the computer, but these should both be included with an up-to-date version of the Raspian operating system. In fact, there will be two versions of Python, which are a more basic but more reliable edition, and the more advanced but less fully debugged Python 3.

These can be run from the GUI where they have icons labelled IDLE (integrated development environment) and IDLE3. However, there will problems when trying to use the GPIO software module if Python is launched from the GUI. The program will not be able to locate the module, and an error message suggesting 'try running from root' will be produced. Python should be launched from the command line. If the computer is set to start with the GUI, run the LXE terminal program from the GUI, and then use its command line to launch Python. These commands are used to launch IDLE and IDLE3 respectively:

```
sudo IDLE
sudo IDLE3
```

For test purposes I ran IDLE3. The program starts in the Python Shell, and here it is possible to use the direct mode. Lines of code that are entered here will be executed immediately. Selecting `New Window` from the `File` menu launches a second window that contains a text editor that is customised for the production of Python programs (Fig.4). Programs can be entered into the text editor and then run in the Python window by selecting `Run Module` from the `Run` menu.

Numbering systems

Initially it is probably best to try a few basic experiments with the program's direct mode. Whether writing a program or using direct mode, the GPIO software module must first be loaded into Python, and the appropriate mode must be set. The two modes are BOARD and BCM, and they offer different methods of numbering when addressing the input output lines. With the BCM method, the channel

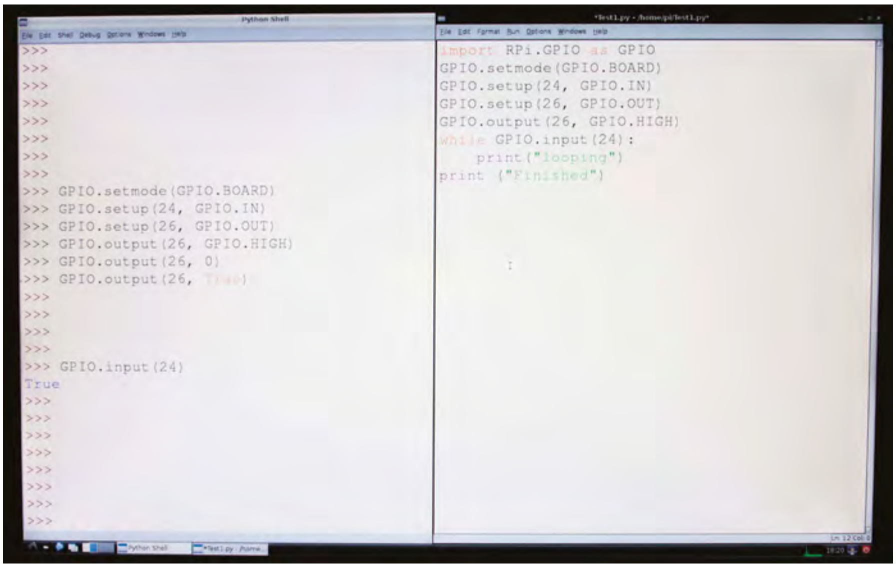


Fig.4. The window on the left contains the Python Shell, and the direct mode can be used here. The window on the right is essentially a text editor, and programs entered here are run via the Shell

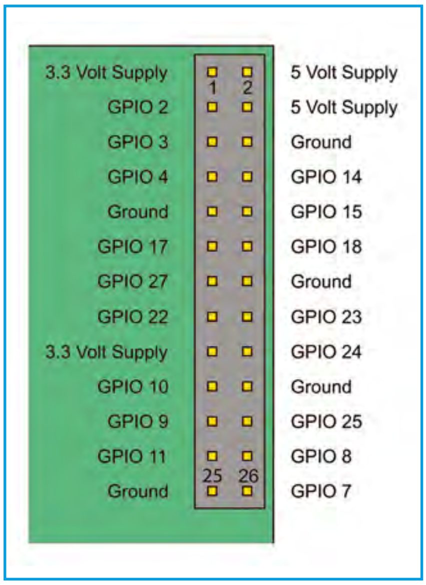


Fig.5. Pin functions and numbering for the GPIO port. The GPIO software add-on can use the channel numbers (BCM mode) or the connector pin numbers (BOARD mode) to access the input/output lines

numbering for the SOC chip at the heart of the Raspberry Pi board is used. With the BOARD method, the numbering of the 26-way GPIO connector is used (see Fig.5). For example, the GPIO 7 input/output line is at pin 26 of the GPIO connector. It would therefore be accessed as line 7 with the BCM mode, or line 26 with the BOARD mode. The BOARD mode seems to be the easier one to use, and it is the mode that will be used here.

These two lines will import the GPIO module and set it to the BOARD mode:

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
```

The first line imports the GPIO module and effectively renames it so that it can be used thereafter as just `GPIO` rather than its full name or `RPi.GPIO`. This seems to be the convention when using the GPIO port. It should be borne in mind that things are generally case sensitive when using any version of the Linux operating system. So, for instance, in the first line it has to be `import`, and not `Import` or `IMPORT`.

Next, the pin or pins that will be used as inputs or outputs must be set to the required mode. For this example, suppose that GPIO 8 at pin 24 is to be set as an input, and that GPIO 7 at pin 26 is to be used as an output. These two lines would have the desired effect:

```
GPIO.setup(24, GPIO.IN)
GPIO.setup(26, GPIO.OUT)
```

When setting a pin as an output it can also be given an initial state, as in this example:

```
GPIO.setup(26,GPIO.OUT,
initial=GPIO.HIGH)
```

This sets pin 26 as an output that is initially high (logic 1). There is no need to set outputs low initially, as this is their default state. It is only fair to point out that when I tried setting an initial output state it failed to work properly with my system. Using a separate instruction to set the initial state of an output would seem to be a more reliable way of doing things. To change the state of pin 26 you would use the appropriate one of these two program lines:

```
GPIO.output(26, GPIO.HIGH)
GPIO.output(26, GPIO.LOW)
```

It is possible to use `1` or `True` in place of `GPIO.HIGH`. Similarly, `0` or `False` can be used instead of `GPIO.LOW`.

Having set pin 24 as an input, it can be read using this line:

```
GPIO.input(24)
```

If you simply wish to print the result of reading the input line there is no need for a print statement, and Python will print the result on screen without using any extra code. This will be `True` if pin 24 is high, or `False` if it is low. Of course, in most cases there will a loop that is controlled by the result of reading the line, or the result will be stored in a variable so that it can be processed later. It is often necessary to have a hold-off until an input line goes to a particular state, and one way of achieving this in Python is to use a while loop, as in this example:

```
while GPIO.input(24):
     print (“looping”)
print (“Finished”)
```

The second line must be indented, as this indicates to Python that it is the code that must be implemented while the condition is met. In this case, it simply prints ‘looping’ repeatedly down the screen while pin 24 is True (high). In practice the instruction used here could be more functional, or it could just be a dummy instruction if nothing more than a hold-off is required. Taking pin 24 `False` (low) ends the loop and `Finished` is printed on the screen by the final line. The parentheses in the print instructions are needed with Python 3, but not with earlier versions.

Nibbling

Using several input lines to read nibbles, bytes, or words of data is very straightforward. Listing 1 reads four-bit nibbles of data using pins 18, 22, 24, and 26, with pin 18 carrying the least-significant bit and pin 26 carrying the most-significant bit. The first six lines simply set everything up with the relevant GPIO pins being set to operate as inputs. The next line sets variable `nibble` at a starting value of zero. There is no need to declare variables in Python programs, and it is therefore perfectly acceptable to simply name one and give it a starting value. The next two lines form an `if` statement that reads pin 18 and then increments `nibble` by one only if this pin is high (`True`). Three more `if` statements then perform the same basic action for pins 22, 24, and 26, but with values of 2, 4, and 8 respectively being added to `nibble` if the relevant input line is high. The value stored in `nibble`, which will be in the range 0 to 15, is then printed on the screen.

Getting the message

The penultimate line of the program clears the setting up that was done by the first six lines of the program. Running another program that uses the GPIO port, or even running the same program again, could otherwise produce messages warning that the hardware is already in use. Setting the GPIO port back to its initial state also ensures that switching to different hardware on this port will not give problems with an output on the port being connected to an output on the other equipment.

```
Listing 1
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(18, GPIO.IN)
GPIO.setup(22, GPIO.IN)
GPIO.setup(24, GPIO.IN)
GPIO.setup(26, GPIO.IN)
nibble = 0
if GPIO.input(18):
     nibble = nibble + 1
if GPIO.input(22):
     nibble = nibble + 2
if GPIO.input(24):
     nibble = nibble + 4
if GPIO.input(26):
     nibble = nibble + 8
print (nibble)
GPIO.cleanup()
print (“Finished”)
```

Some of the pins are set up by the operating system for special purposes, and there can be problems with warning messages being produced if you reassign one of these. It is perfectly all right to reassign pins if they will not be used for their notional purposes, and this program line will suppress unwanted warning messages:

```
GPIO.setwarnings (False)
```

Most practical applications would require 8-bit bytes rather than nibbles to be read, but the program could be easily extended to use all eight lines on the evens side of the GPIO connector. There are seventeen input/output lines in total, so it could even be extended to read 16-bit words. When used as inputs the GPIO lines are quite sophisticated. For example, it is possible to utilise internal pull-up resistors, and transitions on inputs can be detected. It is worth delving into the ‘fine print’ in the GPIO specification in order to fully exploit the possibilities of input lines.

Mastering rotary encoders – Part 1

IN the past few months there have been a couple of threads on *EPE Chat Zone* relating to rotary encoders; ***lincoln*** posted this question:.

I'm using a cheapo (65p by Alps) three-terminal rotary encoder to advance or retard counters in my PIC circuit. One of the outputs interrupts the program and the interrupt then reads the other pin to see if the encoder has been turned forwards or backwards, raising an appropriate flag. Maybe my program has got so big that these flags can't be read and reset quickly enough (3K and counting at 4MHz) but I find the 'action' of the encoder is not as positive as I'd like. When advancing just one click it can actually send the counter the wrong way, and when turning fast it may miss a notch here and there. I started by using a software debounce delay and eventually used a couple of small caps to improve performance, but it's still not ideal.

I'm wondering if a more expensive encoder would give a better (ie sharper, more positive) feel and if so, how does one choose from the hundreds on offer? All the ones below, say £5, look almost the same. Also, none seem to have a reassuring mechanical click like the ones I remember from 10 years ago. They now have a soft detent feel.

Or should I look at the software? Maybe go to 20MHz instead of 4MHz? Maybe I should go for one with fewer clicks per revolution (mine has 24). Any hints greatly appreciated

A little later ***atferrari*** posted:

Optical rotary encoders, as in a PC's mouse, read through two PICs' pins. Schmitt triggers in between – are they convenient, a must or not necessary at all?

This month we will provide an introduction to rotary encoders and next month we will continue with a more detailed look at how to use the signals from incremental encoders, addressing issues such as debouncing, to which lincoln refers.

Angular position measurement

Rotary encoders convert rotational or angular mechanical position or movement into an electronic signal, usually a digital code or sequence. They are examples of position encoders, which convert mechanical position, position change or movement of any form to electronic codes. Rotary encoders are the most commonly encountered position encoders, but other types, such as linear position encoders, are also available.

Rotary encoders have a wide range of uses. Initially, they were mainly found in industrial settings such as machine tools and production line robotics. In these contexts the encoders are often required to meet demanding specifications in terms of precision and robustness, and are therefore expensive, with some examples currently costing several hundred pounds. At the other end of the scale, and more relevant to the *Chat Zone* discussions, low-cost rotary encoders are now often found in the user interfaces of electronic equipment. The cheapest of these cost less than one pound.

Perhaps the most common use of rotary encoders in user interfaces is (or at least was) in mechanical mice and mice scroll wheels (optical mice are now replacing the mechanical variety). In recent years, as more and more equipment is digitally controlled, rotary encoder use has increased and broadened significantly. For example, in the past, volume controls in audio equipment were typically potentiometers (variable resistors) placed directly in the signal path. Now you are likely to find a rotary encoder connected to a microcontroller which in turn sets the volume, either within the digital signal processing, or via a digital potentiometer chip, or similar circuit.

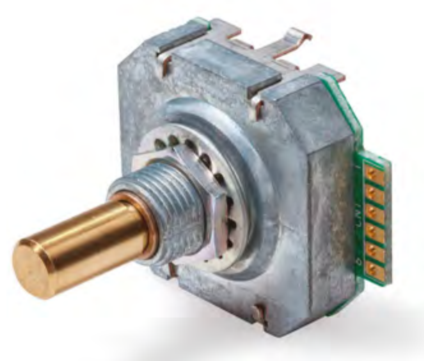
Using a rotary encoder, via a microcontroller, for a volume control is obviously more complex than a simple variable resistor, but has the advantage that the volume setting can be displayed numerically or graphically on an LCD or other display. It also gives the system designers more flexibility in terms of overall control of the system. For example, a moderate default volume can be set at power-up or the previous setting can be recalled as appropriate. Analogue potentiometers sometimes cause problems when used in an audio signal path, particularly the creation of noise when the wiper moves. Using encoders overcomes this, but they are not without potential difficulties – as lincoln's post indicates. We will discuss this in more detail next month.

Of course, volume controls are not the only application of rotary encoders for equipment controls. Any numerical parameter can be adjusted up or down via the software responding to the encoder's movement. The fact that an encoder is controlling the system via software means that a single encoder can be used to adjust many parameters in conjunction with a menu or selection switches (eg, volume, bass and treble for audio). They can also be used to select options, where they take on a role more like a rotary switch than a potentiometer.

Absolute *vs* incremental

There are two types of rotary encoder – absolute and incremental. Absolute encoders output a specific digital code for a specific rotational position. The relationship between the shaft rotary position and output code is fixed by the device's physical structure. Incremental encoders produce signals which indicate the direction and amount of movement, their outputs cannot be interpreted directly as

High quality optical absolute multiturn rotary encoder manufactured by FRABA in Germany. This device is protected to IP66 and has resolution of up to 16 bits (65536 steps) per revolution



Panel-mounting optical incremental rotary encoder available with 16, 24 or 32 detents per revolution

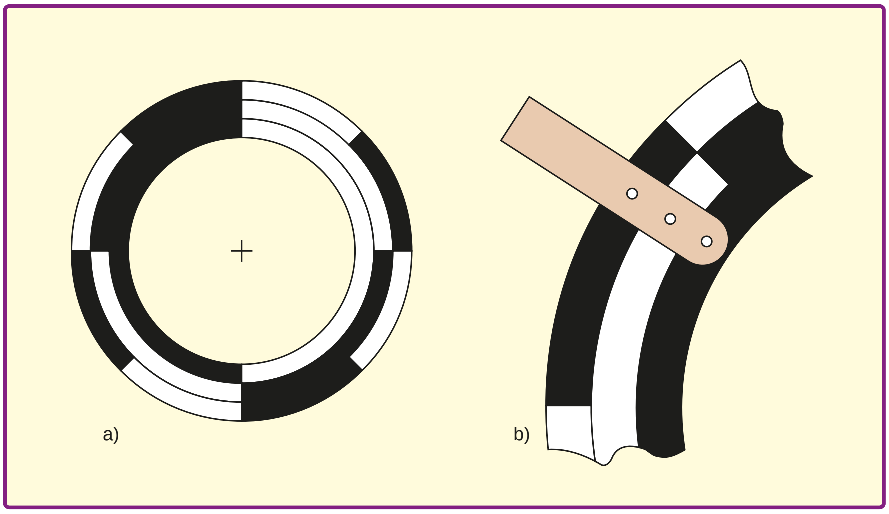


Fig.1 a) Disc patterned with binary code, b) arrangement of contacts/sensors

a specific position – you can only measure relative movement – however, do note that some have an index output in one location which can act as a reference point, potentially allowing absolute position to be tracked.

Absolute encoders provide a position measurement at power up, even if no movement has occurred. Incremental encoders with an index point can only be used to provide a specific position measurement (after power up) after the system returns to the reference location.

Both absolute and incremental encoders are specified in terms of the number of positions they can indicate in a full revolution. There are some multi-turn absolute encoders that can indicate position over multiple revolutions, in which case the number of revolutions measured is also specified. For incremental encoders, unless there is some form of mechanical stop, there is no limit to the number of rotations – the same relative movement outputs are produced indefinitely over multiple revolutions; however, the number of distinct positions that are indicated per revolution is specified.

Absolute encoders tend to be more expensive. The cheapest are a few pounds rather than the sub-pound level of the lowest-cost incremental encoders. The most expensive units for use in machinery tend to be multi-turn absolute encoders.

Incremental encoders are suited for a wide range of user controls. Parameter values (eg, volume) can be increased/decreased over multiple rotations forwards/backwards, irrespective of the number of positions per revolution, as long as the user gets appropriate feedback (via a display, or even just by listening to audio output). However, the amount of rotation needed to produce a step change in a parameter will affect the feel of the control, as indicated in lincoln's comments.

Mechanical feedback – 'feel'

On the subject of 'feel', ***lincoln*** mentions detents. In general, a detent is a mechanism that temporarily holds something in a particular position, which can be released by application of force. Typically, a detent restricts the movement or position of a mechanism to particular directions or positions. Rotary encoders for use as user controls may have detents so that the rotation stops at specific points, which produce stable output signals.

The exact feel/user experience of a particular encoder will depend on its mechanical properties, such as how much of a distinct click any detents produce, as well as the (typically) software-controlled response of the system to user movements. I have not personally evaluated the wide range of available encoders, so I cannot make specific recommendations. However, a quick check of two pieces of nearby audio equipment with digital volume controls revealed one with distinct stepped clicks and the other with a very smooth continuous rotation. Which is 'best' is probably a matter of personal taste.

Encoder technology

Rotary encoders can be implemented in a number of ways, the most common being mechanical and optical. Mechanical encoders use contacts that connect and disconnect as the encoder rotates to produce on/off switching. Optical encoders use LEDs and phototransistors, together with a disc patterned with opaque and transparent sections, or with reflective and non-reflective sections, to produce the required outputs.

Fig.1a shows a disc that could be used for an absolute encoder with eight positions (codes) per revolution. The black and white areas indicate conducting/non-conducting, opaque/transparent or reflective/non-reflective sections, depending on the implementation technology. The positions of the electrical contacts or optical sensors are shown in Fig.1b. As the disk rotates, a different part of the pattern is aligned with the contacts/sensors and so the encoder produces a different output code. The disc in Fig.1a is configured to produce a binary code, as shown in Table 1 below.

This seems fine, but actually it may not be the most sensible way of patterning the disc. The problem is that unless the sensors/contacts are perfectly aligned, as they move from one segment to the next, the bits which change may not all do so at the same instant. This can cause unwanted 'intermediate' codes to be output. The worse case (in our example) is the change from 000 to 111 or vice versa, where any three-bit number (or even a sequence of two numbers) could

Table 1: three-bit binary code from the encoder in Fig.1

A inner ring, Fig.1
B middle ring, Fig.1
C outer ring, Fig.1

Output codes			Position
A	B	C	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Table 2: three-bit Gray code from the encoder in Fig.2

A inner ring, Fig.2
B middle ring, Fig.2
C outer ring, Fig.2

Output codes			Position
A	B	C	
0	0	0	0
0	0	1	1
0	1	1	2
0	1	0	3
1	1	0	4
1	1	1	5
1	0	1	6
1	0	0	7

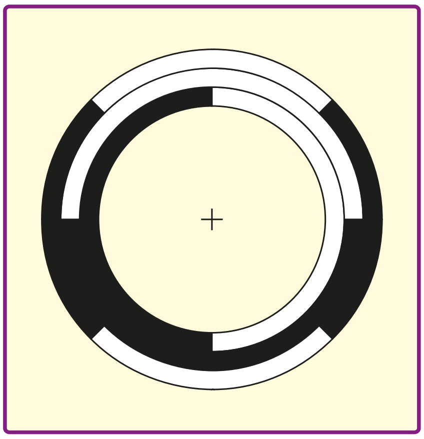


Fig.2 Gray-coded disc

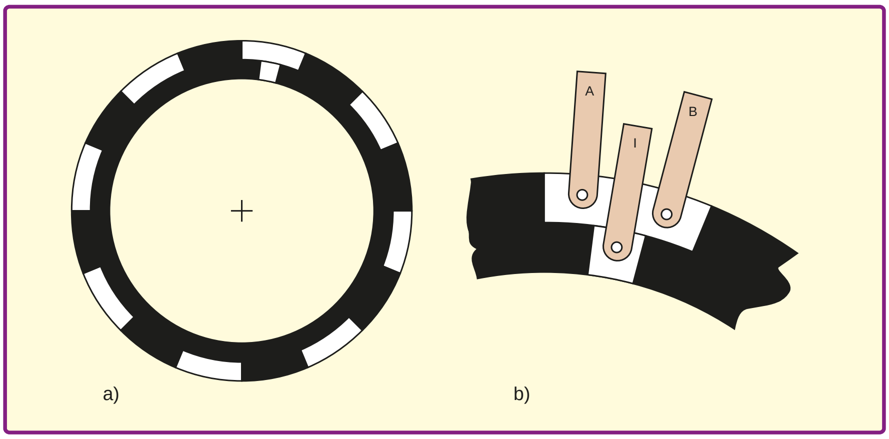


Fig.3 Incremental encoder for motion sensing a) disc pattern, b) sensor locations. Note that many encoders do not have an index output (I)

appear as an intermediate code. If the encoder happened to stop at exactly the wrong location it could potentially produce steady rather than transient erroneous output.

Gray code

The solution is to pattern the disc so that only one bit changes as it moves from one segment to the next, as shown in Fig.2. This type of binary code is called a Gray code and a three-bit example is shown in Table 2. The idea of 'just changing one bit' could produce a number of possible codes, but there is a standard called binary reflected Gray code, which is usually used. We can use a logic circuit or a simple code routine in a microcontroller reading the encoder to convert the Gray code to the corresponding binary numbers representing the disc's position.

If high resolution is required, then the manufacture of an absolute encoder disc becomes complex, requiring very high precision in the patterning and consequently increased cost. Incremental encoders require fewer sensors and less complex patterning of the disc, even for a large number of positions per rotation. The incremental encoder disc pattern and sensor locations are shown in Fig.3. This shows an index location, but this is often not present, particularly on cheap encoders. The spacing of the alternate black and white zones determines the resolution with which location can be measured.

A PCB-mounting continuous incremental encoder with 20 detents per revolution. It uses quadrature output of the encoder to provide direction and pulse. This is typical of the devices used for adjustment controls in domestic appliances

Incremental encoder direction detection

The positioning of the two (main) sensors/switches in the incremental encoder is set up so that they generate a quadrature signal when the disk moves – that is two waveforms offset by a 90° phase shift. The separation distance between sensors/switches A and B is half the length of the black and white sections. The relative position of the sensors/switches means that it is impossible for both of them to change at the same time.

It is possible for one of the sensors to trigger erratically if the encoder happens to stop with one of the sensors/contacts exactly over the switching point. The purpose of the detents mentioned above is to prevent this by ensuring that the resting positions of the encoder sensors/contacts are away from these positions; for example, at a location such as the one depicted in Fig.3(b).

Movement is indicated by the output of either sensor/switch changing, and direction is indicated by the relative phase (±90°) of the signals. These signals can be used to control a hardware or software counter to obtain a binary representation of relative movement. Fig.4 shows waveform of the quadrature signal obtained from sensors/switches A and B in Fig.3.

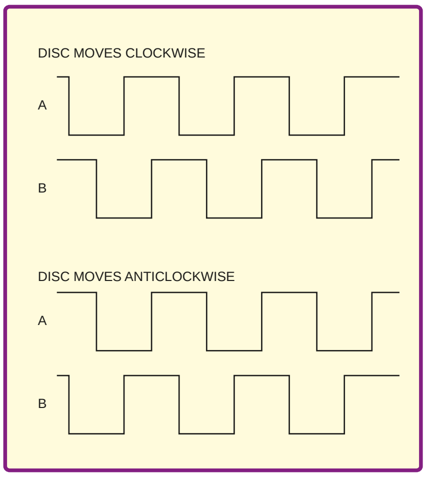


Fig.4 Quadrature signals from the incremental encoder shown in Fig.3. Starting from sensor positions shown in Fig.3. (with logic 0 for black and logic 1 for white)

We can build logic circuits, or write code, to determine the direction of movement from the quadrature signals (A and B in Fig.3 and Fig.4) and increment or decrement a count each time we detect movement. The simplest approach is to look at the level of (say) B when there is (say) a positive edge on A. Such edges can be detected by a flip-flop in hardware or by a 'changed input' interrupt when programming a microcontroller (as lincoln describes). The quadrature signals can also be sampled by a clock signal, with a change between the 'previous' and 'current' values indicating movement.

Checking Fig.4 will show that for a positive edge on A, B=0 indicates clockwise movement, whereas B=1 indicates anticlockwise movement. We can check the direction at any of the edges of the two waveforms in this way. Each edge also represents a definite indication of movement and can therefore be used to increment or decrement a counter circuit depending on the direction. The binary value in the counter will then represent the relative movement of the disc since the counter was last set a particular value (eg, reset). If this reset occurred at a known physical location (eg, via the index sensor in Fig.3) then the absolute position can be determined. We will look at decoding the quadrature signals in more detail next month.

There are a number of issues that may need to be addressed when using incremental encoders for user controls (and other applications). These include the bouncing of mechanical switches, resolution of the position measurement, vibration of the mechanism triggering the sensors/switches, effect of digital noise, and choice between implementation of hardware or software decoding and counting. We will examine some of these issues next month.

Max's Cool Beans

By Max The Magnificent

Exciting times! – I've just got off the phone with a retired engineer who has agreed to sell me an old ASR-33 teletype terminal (also known as a teleprinter, teletypewriter, or TTY). He's had it stored in a warehouse for goodness knows how long. He tells me that this little beauty boasts a keyboard, printer, paper tape reader, paper tape punch, and copy holder, all for $250 (about £160).

So why would I want one of these little beauties, and what am I going to do with it? Well, first of all, let's take a step back to ponder the origin of the teleprinter. Do you think these things were originally intended for use with computers? If so, then I'm afraid you are… how can I put this? … wrong.

The advent of the teleprinter

In 1902, a young American electrical engineer called Frank Pearne was experimenting with printing telegrams – ie, printing messages sent using Morse telegraph. Pearne wanted to develop these machines and was looking for a sponsor. He approached Joy Morton, president of the Morton Salt Company. Morton discussed the situation with his friend, the distinguished mechanical engineer Charles Krum, and they eventually decided they were interested in pursuing this idea.

Pearne left the project to become a teacher, but Krum continued to investigate the problem and, in 1906, was joined by his son Howard, who had recently graduated as an electrical engineer. The mechanical and electrical talents of the Krums complemented each other. After solving the problem of synchronising the transmitter and receiver, they oversaw their first installation on postal lines between New York City and Boston in 1910. These devices, which they called teleprinters, had a typewriter-style keyboard for entering outgoing messages and a roll of paper for printing incoming communications. The Krums continued to improve the reliability of their systems over the years. By 1914, teleprinters were being used by the Associated Press to deliver copy to newspaper offices throughout America, and by the early 1920s they were in general use around the world.

Teleprinters and computers

A decade later in the early 1930s, researchers began to turn their attention to computing. The first devices, such as Vannevar Bush's Differential Analyser, were predominantly analogue, but not everyone was a devotee of analogue computing. In 1937, George Robert Stibitz, a scientist at Bell Laboratories, built a digital machine called the 'Model K', based on relays, flashlight bulbs and metal strips cut from tin-cans ('K' because Stibitz constructed most of it on his *kitchen* table).

Stibitz went on to create a machine called the Complex Number Calculator and, at a meeting in New Hampshire in September 1940, he used this machine to perform the first demonstration of remote computing. Leaving his computer in New York City, he took a teleprinter to the meeting which he connected to the computer using a telephone line. Stibitz then proceeded to astound the attendees by allowing them to pose problems which were entered on the teleprinter; within a minute, the teleprinter printed the answers generated by the computer.

By the 1950s, computers were becoming much more complex, but operators were still largely limited to entering programs using a switch panel, paper tapes or punched cards. Also, because the only way for early computers to be cost-effective was for them to operate twenty-four hours a day, the time-consuming task of writing programs had to be performed off-line using teleprinters with integrated paper tape writers or card punches.

As computers increased in power, teleprinters began to be connected directly to them. This allowed the operators and the computer to communicate directly with each other, which was one of the first steps along the path to the interactive way we use computers today. By the middle of the 1960s, computers had become so powerful that many operators could use the same machine simultaneously, and a new concept called time-sharing was born. The computer could switch between users so quickly that each user had the illusion they had sole access to the machine. (Time-sharing is now practised only in large computing installations, because computers have become so powerful and so cheap that everyone can have a dedicated processor).

Sad to relate (for those of us with a nostalgic bent), the days of the teleprinter in the computing industry were numbered; they were eventually replaced by the combination of computer keyboards and video displays, and the sound of teleprinters chuntering away in computer rooms is now a distant memory.

A roll of drums

And so we return to the ASR-33 teletype terminal, which will soon have pride of place in my office. When I started my career, these little rascals were everywhere. I'm thinking about using a small wireless module to provide a bi-directional communications path between my PC and the ASR-33 (I'll have to mess around with its 20mA current loop interface, but that will only increase the fun). Until next time… have a good one!

ASR-33 teletype terminal with paper tape reader/writer

PICmicro TUTORIALS AND PROGRAMMING

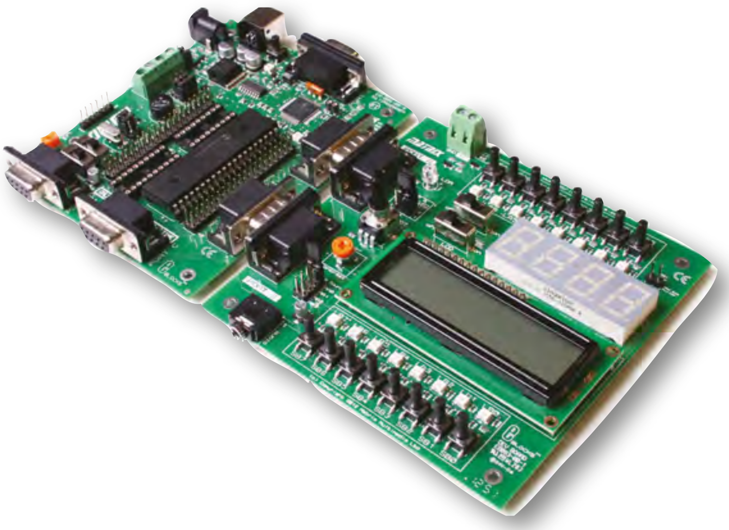
HARDWARE

PICmicro Multiprogrammer Board and Development Board

Suitable for use with the three software packages listed below

This flexible PICmicro microcontroller programmer board and combination board allows students and professional engineers to learn how to program PICmicro microcontrollers as well as program a range of 8, 18, 28 and 40 pin devices from the 12, 16 and 18 series PICmicro ranges. For experienced programmers all programming software is included in the PPP utility that comes with the multiprogrammer board. For those who want to learn, choose one or all of the packages below to use with the hardware.

- Makes it easier to develop PICmicro projects
- Supports low cost Flash-programmable PICmicro devices
- Fully featured integrated displays – 16 individual LEDs, quad 7-segment display and alphanumeric LCD display
- Supports PICmicro microcontrollers with A/D converters
- Fully protected expansion bus for project work
- USB programmable
- Compatible with the E-blocks range of accessories



£167 including VAT and postage, supplied with USB cable and programming software

SOFTWARE

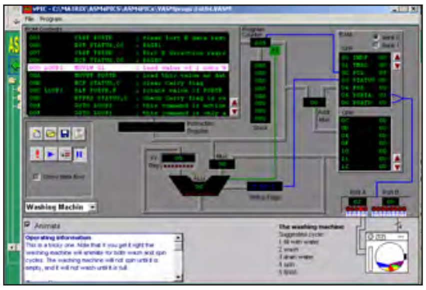
ASSEMBLY FOR PICmicro V4

(Formerly PICtutor)

Assembly for PICmicro microcontrollers V3.0 (previously known as PICtutor) by John Becker contains a complete course in programming the PIC16F84 PICmicro microcontroller from Arizona Microchip. It starts with fundamental concepts and extends up to complex programs including watchdog timers, interrupts and sleep modes.

The CD makes use of the latest simulation techniques which provide a superb tool for learning: the Virtual PICmicro microcontroller, this is a simulation tool that allows users to write and execute MPASM assembler code for the PIC16F84 microcontroller on-screen. Using this you can actually see what happens inside the PICmicro MCU as each instruction is executed, which enhances understanding.

● Comprehensive instruction through 45 tutorial sections ● Includes Vlab, a Virtual PICmicro microcontroller: a fully functioning simulator ● Tests, exercises and projects covering a wide range of PICmicro MCU applications ● Includes MPLAB assembler ● Visual representation of a PICmicro showing architecture and functions ● Expert system for code entry helps first time users ● Shows data flow and fetch execute cycle and has challenges (washing machine, lift, crossroads etc.) ● Imports MPASM files.

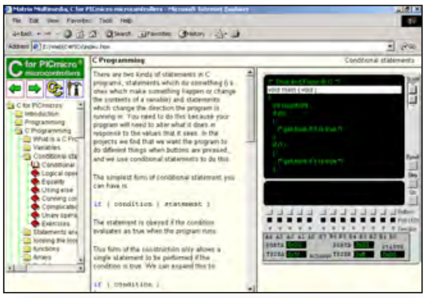


'C' FOR 16 Series PICmicro Version 4

The C for PICmicro microcontrollers CD-ROM is designed for students and professionals who need to learn how to program embedded microcontrollers in C. The CD-ROM contains a course as well as all the software tools needed to create Hex code for a wide range of PICmicro devices – including a full C compiler for a wide range of PICmicro devices.

Although the course focuses on the use of the PICmicro microcontrollers, this CD-ROM will provide a good grounding in C programming for any microcontroller.

● Complete course in C as well as C programming for PICmicro microcontrollers ● Highly interactive course ● Virtual C PICmicro improves understanding ● Includes a C compiler for a wide range of PICmicro devices ● Includes full Integrated Development Environment ● Includes MPLAB software ● Compatible with most PICmicro programmers ● Includes a compiler for all the PICmicro devices.



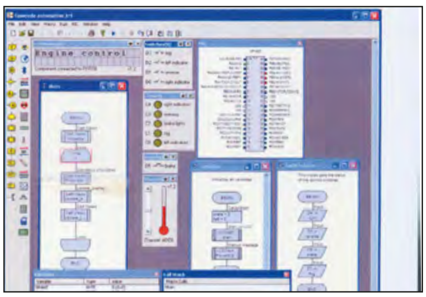
Minimum system requirements for these items: Pentium PC running, 2000, ME, XP; CD-ROM drive; 64MB RAM; 10MB hard disk space.
Flowcode will run on XP or later operating systems

FLOWCODE FOR PICmicro V5 (see opposite page)

Flowcode is a very high level language programming system based on flowcharts. Flowcode allows you to design and simulate complex systems in a matter of minutes. A powerful language that uses macros to facilitate the control of devices like 7-segment displays, motor controllers and LCDs. The use of macros allows you to control these devices without getting bogged down in understanding the programming. When used in conjunction with the Version 3 development board this provides a seamless solution that allows you to program chips in minutes.

- Requires no programming experience
- Allows complex PICmicro applications to be designed quickly
- Uses international standard flow chart symbols
- Full on-screen simulation allows debugging and speeds up the development process.
- Facilitates learning via a full suite of demonstration tutorials
- Produces ASM code for a range of 18, 28 and 40-pin devices
- 16-bit arithmetic strings and string manipulation
- Pulse width modulation
- I2C.

Features include panel creator, in circuit debug, virtual networks, C code customisation, floating point and new components. The Hobbyist/Student version is limited to 4K of code (8K on 18F devices)



PRICES

Prices for each of the CD-ROMs above are:

(Order form on next page)

(UK and EU customers add VAT to 'plus VAT' prices)

Hobbyist/Student	£58.80	inc VAT
Professional (Schools/HE/FE/Industry)	£150	plus VAT
Professional 10 user (Network Licence)	£499	plus VAT
Site Licence	£999	plus VAT
Flowcode Professional (Schools/HE/FE/Industry)	£199	plus VAT
Flowcode 10 user (Network Licence)	£599	plus VAT
Flowcode Site Licence	£999	plus VAT

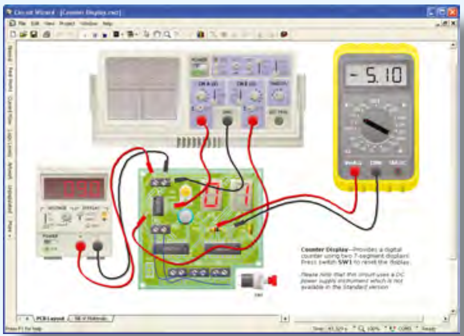
CIRCUIT WIZARD

Circuit Wizard is a revolutionary software system that combines circuit design, PCB design, simulation and CAD/CAM manufacture in one complete package.

Two versions are available, Standard or Professional.

By integrating the entire design process, Circuit Wizard provides you with all the tools necessary to produce an electronics project from start to finish – even including on-screen testing of the PCB prior to construction!

✱ Circuit diagram design with component library (500 components Standard, 1500 components Professional) ✱ Virtual instruments (4 Standard, 7 professional) ✱ On-screen animation ✱ Interactive circuit diagram simulation ✱ True analogue/digital simulation ✱ Simulation of component destruction ✱ PCB Layout ✱ Interactive PCB layout simulation ✱ Automatic PCB routing ✱ Gerber export ✱ Multi-level zoom (25% to 1000%) ✱ Multiple undo and redo ✱ Copy and paste to other software ✱ Multiple document support



This software can be used with the *Jump Start* and *Teach-In 2011* series (and the *Teach-In 4* book).

Standard **£61.25** inc. VAT Professional **£91.90** inc. VAT

Minimum system requirements for these CD-ROMs: Pentium PC, CD-ROM drive, 32MB RAM, 10MB hard disk space. Windows 2000/ME/XP, mouse, sound card, web browser.

GCSE ELECTRONICS

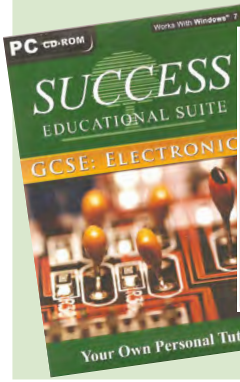
Suitable for any student who is serious about studying and who wants to achieve the best grade possible. Each program's clear, patient and structured delivery will aid understanding of electronics and assist in developing a confident approach to answering GCSE questions. The CD-ROM will be invaluable to anyone studying electronics, not just GCSE students.

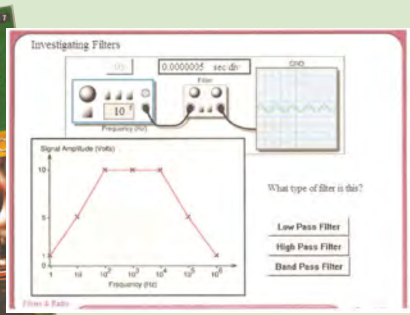
✱ Contains comprehensive teaching material to cover the National Curriculum syllabus ✱ Regular exercises reinforce the teaching points ✱ Retains student interest with high quality animation and graphics ✱ Stimulates learning through interactive exercises ✱ Provides sample examination ques-tions with model solutions ✱ Authored by practising teachers ✱ Covers all UK examination board syllabuses ✱ Caters for all levels of ability ✱ Useful for self-tuition and revision

SUBJECTS COVERED

Electric Circuits – Logic Gates – Capacitors & Inductors – Relays – Transistors – Electric Transducers – Operational Amplifiers – Radio Circuits – Test Instruments

Over 100 different sections under the above headings





£12.50 inc. VAT and P&P

Please send me: **CD-ROM ORDER FORM**

- ☐ Assembly for PICmicro V4
- ☐ 'C' for 16 Series PICmicro V4
- ☐ Flowcode for PICmicro V5 (DOWNLOAD + CDROM)
- ☐ Flowcode for PICmicro V5 (DOWNLOAD ONLY)
- ☐ Flowcode for AVR V5 (DOWNLOAD + CDROM)
- ☐ Flowcode for AVR V5 (DOWNLOAD ONLY)
- ☐ Flowcode for ARM V5 (DOWNLOAD + CDROM)
- ☐ Flowcode for ARM V5 (DOWNLOAD ONLY)
- ☐ Flowcode for dsPIC V5 (DOWNLOAD + CDROM)
- ☐ Flowcode for dsPIC V5 (DOWNLOAD ONLY)
- ☐ Flowcode for PIC24 V5 (DOWNLOAD + CDROM)
- ☐ Flowcode for PIC24 V5 (DOWNLOAD ONLY)
- ☐ Flowkit

Maestro MasterCard VISA

Version required:
- ☐ Hobbyist/Student
- ☐ Professional
- ☐ Professional 10 user
- ☐ Site licence

Email: . . . .

Username: . . . .

Note: The software on each version is the same, only the licence for use varies.

☐ **PICmicro Multiprogrammer Board and Development Board V4 (hardware)**

- ☐ Circuit Wizard – Standard
- ☐ Circuit Wizard – Professional
- ☐ GCSE Electronics

Full name: . . . .

Address: . . . .

. . . . Post code: . . . . Tel. No: . . . .

Signature: . . . .

☐ I enclose cheque/PO in £ sterling payable to WIMBORNE PUBLISHING LTD for £ . . . .

☐ Please charge my Visa/Mastercard/Maestro: £ . . . .

Valid From: . . . . Card expiry date: . . . .

Card No: . . . . Maestro Issue No. . . . .

Card Security Code . . . . (The last 3 digits on or just under the signature strip)

ORDERING
ALL PRICES INCLUDE UK POSTAGE

Student/Single User/Standard/Hobbyist Version price includes postage to most countries in the world
EU residents outside the UK add £5 for airmail postage per order

Professional, Multiple User and Site License Versions – overseas readers add £5 to the basic price of each order for airmail postage (**do not add VAT** unless you live in an EU (European Union) country, then add VAT at 20% or provide your official VAT registration number).

Send your order to:
Direct Book Service
Wimborne Publishing Ltd
113 Lynwood Drive, Merley,
Wimborne, Dorset BH21 1UU

To order by phone ring

01202 880299. Fax: 01202 843233

Goods are normally sent within seven days

E-mail: orders@wimborne.co.uk

Online shop:

www.epemag.com

NET WORK

by Alan Winstanley

Google Googles

WHEN my industrial career was in its infancy, a colleague made a comedy routine out of wearing lens-free glasses, sometimes poking his finger through the open frames to rub his eyes, much to the bemusement of onlookers and creating ripples of mirth around the office. This brings me to the much anticipated release of Google's next generation mobile Internet gadget, namely Google Glass, which I mentioned last month. This pair of sci-fi specs. bears a tiny projector like a small block of clear acrylic that displays a tiny screen before the wearer's eyeball. It enables the wearer to check their phone through Bluetooth, access the web or connect to Wi-Fi, check navigation directions on a head-up display or interact with its built-in camera.

The early examples of bare-bones Google Glass consist of a lens-free lightweight frame with the eyepiece on one side, with clear and smoked visors included separately: you can try to wear Google Glass over existing spectacles or sport them on their own, bare-framed and without a visor. The frame can be tapped or swiped to perform basic actions and voice-activated operations can also be carried out. The device rests on the nose like any pair of lens-less spectacles would, and non-spectacles wearers should be able to view the 720p display directly in the plastic prism, like a head-up display above the natural line of sight. Sound is delivered through a bone-conducting transducer and battery power is said to be in the low number of hours, in keeping with the gadget's compact dimensions.

Some testers say that Google Glass doesn't do all that much, with one user commenting that in some respects it's like using a last-generation smartphone that is dedicated firmly to a Google account. Currently, Google Glass has a high novelty value, as did those embarrassing Bluetooth earpieces when they first appeared, but would you want to wear a pair of Google Goggles, and how safe and secure would you be when doing so? Just like Google Streetview, Google Glass may also provoke a controversy surrounding privacy and photographing subjects without their knowledge or consent.

Some positive applications for Google Glass spring to mind though, and it's already being mooted that they could find their way into spectator sports such as NFL football, offering a player's eye view of the action or perhaps the referee could wear a pair and share his viewpoint with restless crowds. Maybe movie clips will stream directly from Youtube (also owned by Google) onto large screens to give soccer crowds a point-of-view experience like they've never had before.

Early demos of Google Glass also suggested sky-diving and outbound trekking, which would make the most of Google navigation, aided by data fed through a mobile phone. In the UK and some US states the wearing of Google Glasses while behind the wheel of a car is already in the process of being outlawed: an uncharacteristic pre-emptive legal strike before the product has even arrived on our shores. Expect more news of this interesting and possibly controversial product in 2014.

HTC's One for all

For traditional Internet users weaned on dial-up networking, the home computer has been at the heart of everyday Internet activity for two decades or more. Surfing and emailing using a keyboard and mouse seems productive and rewarding, data is backed up slavishly onto disk and hard copies are dutifully printed. Software is patched or upgraded, and new programs are loaded via CD or installed off the web. DVDs are watched or burned, then the Wi-Fi drops out, programs crash and you find that software serial numbers are lost or the software disk is scratched! You might plug in a memory stick and print off some photos, or sort out some online shopping or banking chores. The kids' homework gets googled and essays sprinkled with clip-art get printed, and a webcam brings distant friends and family together. Family trees are researched. In short, a PC or Mac is an indispensable workhorse at the hub of many homes.

The downside is that much of the bulky and wasteful equipment has been tethered to an Ethernet cable for too long, and it takes up valuable space in a corner; it's power hungry (needing 26 sockets in my case), the evolution of ADSL and wireless networking has been tortuous, home Wi-Fi networking can be frustrating and the total cost of ownership is high in terms of material costs, bug fixes, updates and general hair-pulling when things go wrong. We will probably always need a home PC network to tackle the typical domestic 'donkey work', but mobile access has become the dominant feature on the digital landscape. Such is the pace of development that Internet users are increasingly pressured to either embrace it or risk being run over by the mobile revolution.

Keep taking the tablets

Apart from laptops and notebooks, mobile Internet usage comes in other forms, including touch-screen smartphones, tablets or tablets with smartphone capabilities (so-called 'phablets') such as the forthcoming Samsung Galaxy Tab 3. I find it handy to have a cheap tablet kicking around on the sofa sometimes, in order to check email or maybe do a little surfing. More demanding or considered work means heading over towards my PC though.

The HTC One is machined from a billet of aluminium, and is optimised for mobile Internet usage

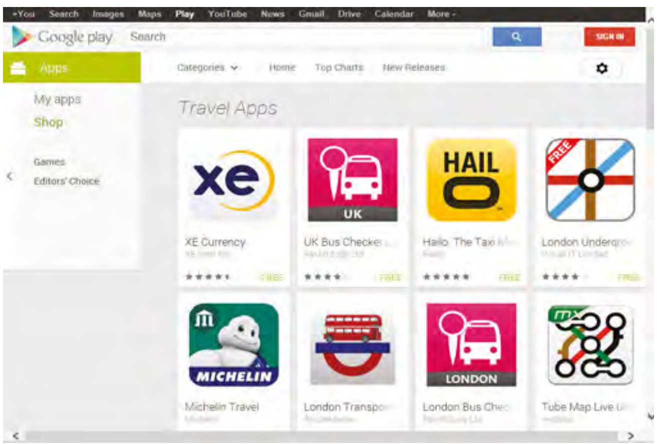
Wanting to do more work when away from my desk, I finally replaced my aging HTC Tytn II first-generation smartphone. The Windows Mobile 6.1 device was well past its prime and had been repaired several times, helped by spares scavenged from eBay. I was also missing out on the many benefits that mobile networking now brings, as I'll explain, and it also interfered like crazy with anything that had a loudspeaker attached!

Mobile users face a bewildering variety of products and packages to pick from, and price comparison sites such as uSwitch can pull together the best mobile phone deals. Choices for new smartphones should start with their operating systems, and they currently boil down to Apple's expensive iPhone, any number of Android phones from Samsung (notably the high-end Galaxy S4), HTC, Huawei, LG, Sony and more, a smaller selection of high quality Windows 8 Mobile devices (especially from Nokia) or Blackberrys. Google has its eye on the mobile phone market too: it already markets the likeable Nexus phone and tablet (a new Nexus 2 tablet is due imminently); Motorola is now a Google company and its newest phone, the Moto X, is very eagerly anticipated but may not make it to the UK.

High speed 3G and 4G network coverage is still thin on the ground, but mobile data packages have to be considered closely too. A tariff such as 500Mb or 1Gb per month of data may be included at a flat rate. Large updates, video calls or downloads are best handled via Wi-Fi, but in any case I have often struggled to even scratch the surface of my monthly 1Gb data allowance. Wi-Fi takes over automatically at home or in the office, or you might find a Wi-Fi hotspot to hook into, so you might not need as large a data bundle as you might think. However, do guard against roaming costs while overseas, as this can clock up an eyewatering bill.

I greatly preferred the build quality of the HTC One phone – machined from a solid aluminum billet – over its lightweight South Korean counterparts, so I chose that model, running the current Android 4.3 (Jelly Bean). Android 5 (Key Lime Pie) is due later this year but has yet to break cover. In my case, Tesco (which runs the O2 mobile network) offered a transparent Anytime Upgrade contract with the handset cost shown separately: once the phone is paid for, the bill drops to just £6.50 a month to cover calls and data. I opted to cap the bill so the monthly costs cannot be exceeded due to excessive usage or theft.

After some bothersome problems with Tesco Mobile logins not working, my new phone promptly arrived by courier. In due course I'll show some of the surprising tricks that a typical Android phone has up its sleeve, which will perhaps inspire readers to get more from home networking or the Internet. The same principles also apply to a decent Android tablet, and an army of them is sold on eBay or direct from China. If you don't want to buy a new phone, you can experience Android on a cheap tablet instead. For example **ahappydeal.com** or **goodluckbuy.com** are well established and sell a vast range of electronics, including Android (Jelly Bean) tablets starting at £30: in the UK you should beware of 20% VAT and import duty though. Fixed fees (brokerage) have an impact on smaller orders so try to bulk up shipments to dilute the cost.



Visit play.google.com to view the latest apps for Android devices

There's an app for that

The dynamics of Internet usage are changing fast, and it's now routine to use a mobile phone to check email or surf the web without a second thought, or download a QA barcode that acts as an airline check-in docket. Compare the prices of competitors while browsing a retail store (the cheek of it!) or tweet in real-time about a great (or not so great) restaurant meal, or upload a photo of your dinner. Book flights online and keep friends on Facebook posted with photos of your overseas holiday, thanks to the hotel's free Wi-Fi, and monitor your home at the same time using an IP webcam with remote control. Find your way around using Google Maps, and even step off the street inside some places and wander round, thanks to the extraordinary interior photography produced for Google Business Photos by Local Exposure (**www.localexposure.co.uk**). Hobbyists are also rigging Raspberry Pi's to the Internet to control anything from aerial photography to remote doorbells over the web.

If yesterday we talked of 'cyberspace' or the 'information superhighway', then today's moniker is probably 'there's an app for that'. A vast choice of these small applications, games and utility programs can be downloaded for just a few pounds or less. Apple's apps are rigorously controlled and are of consistently high quality, but unlike Apple's mobile iOS or Microsoft Windows, Google's highly popular Android operating system is open-source (see: **http://source.android.com**) with its underpinnings built on Linux, making the Android platform widely accessible to developers and manufacturers. Hence a very wide range of Android apps is available, which can be accessed through Google Play (a Google account is needed), but they can be of varying quality. Apps are even being launched for Google Glass wearers. A major bone of contention for Windows Mobile users is that a more restricted choice of Windows apps. is available, but many Windows users don't mind and they luxuriate in the high quality photos taken by their latest Nokia camera phones instead.

My new phone has just beeped, with a chaser email from *EPE* asking for this month's *Net Work* copy! In next month's column I'll explain how I got on when migrating to a new generation Android phone, and I will highlight what a typical smartphone can do for *Net Work* readers.

Basic Soldering Guide on Kindle

Last, this month, readers, allow me to introduce the new Amazon Kindle version of my *Basic Soldering Guide*. The all-new ebook is produced in association with UK soldering iron manufacturer Antex and contains over 80 new photos explaining all the preparation and steps needed to solder electronics successfully. It has been extensively re-written and includes much handy information on soldering techniques that will get beginners up and running in no time. I took on board readers' feedback and questions posed over many years and tackled a few subjects such as Colophony flux, making Western Union joints and plenty more besides. You don't need a Kindle device to read it, as free readers are available for popular platforms. *The Basic Soldering Guide* is out now on Amazon.

That's all for this month's *Net Work*. You can contact the author at **alan@epemag.demon.co.uk** or write to the editor at **editorial@wimborne.co.uk** for possible inclusion in *Readout*, and you could win a valuable prize!

The author's all-new Basic Soldering Guide *for Kindle is written in association with Antex (Electronics). Free readers and mobile apps for PC, Mac and smartphones are also available*

DIRECT BOOK SERVICE

ELECTRONICS TEACH-IN BUNDLE – SPECIAL BUNDLE PRICE £14 FOR PARTS 1, 2 & 3

FREE CD-ROM

ELECTRONICS TEACH-IN 2 CD-ROM
USING PIC MICROCONTROLLERS A PRACTICAL INTRODUCTION

This *Teach-In* series of articles was originally published in *EPE* in 2008 and, following demand from readers, has now been collected together in the *Electronics Teach-In 2* CD-ROM.

The series is aimed at those using PIC microcontrollers for the first time. Each part of the series includes breadboard layouts to aid understanding and a simple programmer project is provided.

Also included are 29 *PIC N' Mix* articles, also republished from *EPE*. These provide a host of practical programming and interfacing information, mainly for those that have already got to grips with using PIC microcontrollers. An extra four part beginners guide to using the C programing language for PIC microcontrollers is also included.

The CD-ROM also contains all of the software for the *Teach-In 2* series and *PIC N' Mix* articles, plus a range of items from Microchip – the manufacturers of the PIC microcontrollers. The material has been compiled by Wimborne Publishing Ltd. with the assistance of Microchip Technology Inc.

CD-ROM **Order code ETI2 CD-ROM** **£9.50**

ELECTRONICS TEACH-IN 3

The three sections of this book cover a very wide range of subjects that will interest everyone involved in electronics, from hobbyists and students to professionals. The first 80-odd pages of Teach-In 3 are dedicated to *Circuit Surgery*, the regular *EPE* clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers the practical aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and indentifying components, are covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by the readers of *EPE*.

The free cover-mounted CD-ROM is the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge, TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The Teach-In 1 series covers everything from Electric Current through to Microprocessors and Microcontrollers and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

160 pages **Order code ETI3** **£8.50**

Book and CD-ROMs **Order code ETIBUNDLE** Bundle Price **£14.00**

ELECTRONICS TEACH-IN 4

FREE CD-ROM

ELECTRONICS TEACH-IN 4

A Broad-Based Introduction to Electronics plus FREE CD-ROM

The Teach-In 4 book covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the new Diploma in Engineering, Level 2.

The Free cover-mounted CD-ROM contains the full Modern Electronics Manual, worth £29.95. The Manual contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to all those interested in learning about electronics or brushing up on their theory, be they hobbyists, students or professionals.

144 pages **Order code ETI4** **£8.99**

RASPBERRY PI

RASPBERRY Pi MANUAL: A practical guide to the revolutionary small computer

176 pages **Order code H001** **£17.99**

RASPBERRY Pi USER-GUIDE

262 pages **Order code JW001** **£12.99**

PROGRAMMING THE RASPBERRY Pi

192 pages **Order code MGH4** **£10.99**

NEW

ELECTRONICS TEACH-IN 5

FREE CD-ROM

ELECTRONICS TEACH-IN 5

Jump Start – 15 design and build circuit projects dedicated to newcomers or those following courses in school and colleges. The projects are: Moisture Detector, Quiz Machine, Battery Voltage Checker, Solar-Powered Charger, Versatile Theft Alarm, Spooky Circuits, Frost Alarm, Mini Christmas Lights, iPod Speaker, Logic Probe, DC Motor Controller, Egg Timer, Signal Injector Probe, Simple Radio Receiver, Temperature Alarm.

PLUS:

PIC' N MIX – starting out with PIC Microcontrollers and PRACTICALLY SPEAKING – the techniques of project construction.

FREE CD-ROM – The free CD-ROM is the complete Teach-In 2 book providing a practical introduction to PIC Micrprocessors plus MikroElektronika, Microchip and L-Tek PoScope software.

160 Pages **Order code ETI5** **£8.99**

MICROPROCESSORS

INTERFACING PIC MICROCONTROLLERS
Martin Bates

298 pages **Order code NE48** **£30.99**

PROGRAMMING 16-BIT PIC MICROCONTROLLERS IN C – LEARNING TO FLY THE PIC24 Lucio Di Jasio (Application Segments Manager, Microchip, USA)

496 pages +CD-ROM **Order code NE45** **£38.00**

INTRODUCTION TO MICROPROCESSORS AND MICROCONTROLLERS – SECOND EDITION
John Crisp

222 pages **Order code NE31** **£29.99**

THE PIC MICROCONTROLLER YOUR PERSONAL INTRODUCTORY COURSE – THIRD EDITION. John Morton

270 pages **Order code NE36** **£25.00**

PIC IN PRACTICE (2nd Edition)
David W. Smith

308 pages **Order code NE39** **£24.99**

MICROCONTROLLER COOKBOOK
Mike James

240 pages **Order code NE26** **£36.99**

The books listed have been selected by *Everyday Practical Electronics* editorial staff as being of special interest to everyone involved in electronics and computing. They are supplied by mail order direct to your door. Full ordering details are given on the next page.

FOR A FULL DESCRIPTION OF THESE BOOKS AND CD-ROMS SEE THE SHOP ON OUR WEBSITE www.epemag.com

All prices include UK postage

1

PROJECT CONSTRUCTION

IC 555 PROJECTS
E. A. Parr

167 pages **Order code BP44** **£5.49**

ELECTRONIC PROJECT BUILDING FOR BEGINNERS R. A. Penfold

135 pages **Order code BP392** **£5.99**

PRACTICAL FIBRE-OPTIC PROJECTS
R. A. Penfold

132 pages **Order code BP374** **£5.45**

THEORY AND REFERENCE

PRACTICAL ELECTRONICS HANDBOOK – Fifth Edition. Ian Sinclair

440 pages **Order code NE21** **£32.50**

BEBOP TO THE BOOLEAN BOOGIE
Second Edition Clive (call me Max) Maxfield

470 pages **Order code BEB1** **£35.99**

BEBOP BYTES BACK (and the Beboputer Computer Simulator) CD-ROM
Clive (Max) Maxfield and Alvin Brown

Over 800 pages in Adobe Acrobat format

CD-ROM **Order code BEB2 CD-ROM** **£21.95**

STARTING ELECTRONICS – 4th Edition
Keith Brindley

296 pages **Order code NE100** **£18.99**

ELECTRONIC CIRCUITS – FUNDAMENTALS & APPLICATIONS – Third Edition Mike Tooley

400 pages **Order code NE43** **£25.99**

FUNDAMENTAL ELECTRICAL AND ELECTRONIC PRINCIPLES – Third Edition C.R. Robertson

368 pages **Order code NE47** **£21.99**

A BEGINNER'S GUIDE TO TTL DIGITAL ICs
R.A. Penfold

142 pages **Order code BP332** **£5.45**

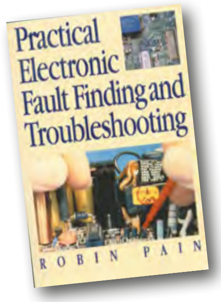
UNDERSTANDING ELECTRONIC CONTROL SYSTEMS
Owen Bishop

228 pages **Order code NE35** **£36.99**

FAULT FINDING AND TEST EQUIPMENT

GETTING THE MOST FROM YOUR MULTIMETER
R. A. Penfold
96 pages Order code BP239 £5.49

PRACTICAL ELECTRONIC FAULT FINDING AND TROUBLESHOOTING
Robin Pain
274 pages Order code NE22 £41.99



HOW ELECTRONIC THINGS WORK – AND WHAT TO DO WHEN THEY DON'T
Robert Goodman
394 pages Order code MGH3 £21.99

OSCILLOSCOPES – FIFTH EDITION
Ian Hickman
288 pages Order code NE37 £36.99

HOW TO USE OSCILLOSCOPES AND OTHER TEST EQUIPMENT
R.A. Penfold
104 pages Order code BP267 £5.49

AUDIO & VIDEO

VALVE AMPLIFIERS – Second Edition
Morgan Jones
288 pages Order code NE33 £40.99

BUILDING VALVE AMPLIFIERS
Morgan Jones
368 pages Order code NE40 £29.00

DIGITAL AUDIO RECORDING
Ian Waugh
60 pages Order code PC121 £7.95

QUICK GUIDE TO MP3 AND DIGITAL MUSIC
Ian Waugh
60 pages Order code PC119 £7.45

MAKING MUSIC WITH YOUR COMPUTER
Stephen Bennett
92 pages Order code PC120 £10.95

QUICK GUIDE TO DIGITAL AUDIO RECORDING
Ian Waugh
208 pages Order code PC121 £7.95

ELECTRONIC PROJECTS FOR VIDEO ENTHUSIASTS
R.A. Penfold
109 pages Order code BP356 £5.45

COMPUTING AND ROBOTICS

NEWNES INTERFACING COMPANION
Tony Fischer-Cripps
295 pages Order code NE38 £41.00

HOW TO BUILD A COMPUTER MADE EASY
R.A. Penfold
120 pages Order code BP707 £8.49

EASY PC CASE MODDING
R.A. Penfold
192 pages + CDROM Order code BP542 £8.99

FREE DOWNLOADS TO PEP-UP AND PROTECT YOUR PC
R.A. Penfold
128 pages Order code BP722 £7.99

WINDOWS XP EXPLAINED
N. Kantaris and P.R.M. Oliver
264 pages Order code BP514 £7.99

THE INTERNET – TWEAKS, TIPS AND TRICKS
R. A. Penfold
128 pages Order code BP721 £7.99

eBAY – TWEAKS, TIPS AND TRICKS
R. A. Penfold
128 pages Order code BP716 £7.50

AN INTRODUCTION TO eBAY FOR THE OLDER GENERATION
Cherry Nixon
120 pages Order code BP709 £8.49

COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
308 pages Order code BP601 £8.99

ANDROIDS, ROBOTS AND ANIMATRONS
Second Edition – John Iovine
224 pages Order code MGH1 £16.99

ROBOT BUILDERS COOKBOOK
Owen Bishop
366 pages Order code NE46 £26.00

INTRODUCING ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
288 pages + Order code BP901 £14.99

MORE ADVANCED ROBOTICS WITH LEGO MINDSTORMS – Robert Penfold
298 pages Order code BP902 £14.99

HOW TO FIX YOUR PC PROBLEMS
R. A. Penfold
128 pages Order code BP705 £8.49

WINDOWS 7 – TWEAKS, TIPS AND TRICKS
Andrew Edney
120 pages Order code BP708 £8.49

GETTING STARTED IN COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
120 pages Order code BP704 £8.49

BOOK ORDERING DETAILS

All prices include UK postage. For postage to Europe (air) and the rest of the world (surface) please add £3 per book. Surface mail can take up to 10 weeks to some countries. For the rest of the world airmail add £4 per book. CD-ROM prices include VAT and/or postage to anywhere in the world. Send a PO, cheque, international money order (£ sterling only) made payable to **Direct Book Service** or card details, Visa, Mastercard or Maestro to: **DIRECT BOOK SERVICE, WIMBORNE PUBLISHING LIMITED, 113 LYNWOOD DRIVE, MERLEY, WIMBORNE, DORSET BH21 1UU**.

Books are normally sent within seven days of receipt of order, but please allow 28 days for delivery – more for overseas orders. *Please check price and availability (see latest issue of* Everyday Practical Electronics*) before ordering from old lists.*

For a full description of these books please see the shop on our website.
Tel 01202 880299 Fax 01202 843233. E-mail: dbs@wimborne.co.uk

Order from our online shop at: www.epemag.com

BOOK ORDER FORM

Full name: ....

Address: ....

....

....

.... Post code: .... Telephone No: ....

Signature: ....

☐ I enclose cheque/PO payable to DIRECT BOOK SERVICE for £ ....

☐ Please charge my card £ .... Card expiry date..

Card Number .... Maestro Issue No............................

Valid From Date .... Card Security Code .... (The last three digits on or just below the signature strip)

Please send book order codes: ....

....

Please continue on separate sheet of paper if necessary

PCB SERVICE

CHECK US OUT ON THE WEB

Basic printed circuit boards for most recent *EPE* constructional projects are available from the *PCB Service*, see list. These are fabricated in glass fibre, and are drilled and roller tinned, but all holes are a standard size. They are not silk-screened, nor do they have solder resist. Double-sided boards are **NOT plated through hole** and will require 'vias' and some components soldering to both sides. **NOTE: PCBs from the July 2013 issue with eight digit codes** have silk screen overlays and, where applicable, are double-sided, plated through-hole, with solder masks, they are similar to the photos in the relevent project articles.

All prices include VAT and postage and packing. Add £2 per board for airmail outside of Europe. Remittances should be sent to **The PCB Service, Everyday Practical Electronics, Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU. Tel: 01202 880299; Fax 01202 843233; Email: orders@epemag.wimborne.co.uk. On-line Shop: www.epemag.com**. Cheques should be crossed and made payable to *Everyday Practical Electronics* **(Payment in £ sterling only)**.

NOTE: While 95% of our boards are held in stock and are dispatched within seven days of receipt of order, please allow a maximum of 28 days for delivery – overseas readers allow extra if ordered by surface mail.

Back numbers or photocopies of articles are available if required – see the Back Issues page for details. WE DO NOT SUPPLY KITS OR COMPONENTS FOR OUR PROJECTS.

PROJECT TITLE	ORDER CODE	COST
SEPTEMBER '12		
Hearing Loop Receiver	865	£8.55
★ Ultrasonic Anti-Fouling For Boats	866	£9.14
Jump Start – Versatile Theft Alarm	867	£9.33
OCTOBER '12		
S/PDIF To TOSLINK Converter	868	£8.16
TOSLINK to S/PDIF Converter	869	£8.16
★ Digital Lighting Controller		
– Master Board	870	£12.05
– Slave Board	871	£16.72
Jump Start – Crazy Eyes	872	£7.78
– Ghostly Sounds	873	£8.16
NOVEMBER '12		
Hearing Loop Level Meter	874	£9.53
RFID Security System	875	£7.75
Jump Start – Frost Alarm	876	£8.55
DECEMBER '12		
Hot Wire Cutter – Controller	877	£8.55
★ Universal USB Data Logger – Part 1 (double-sided)	878	£16.52
Jump Start – Mini Christmas Lights	879	£10.69
JANUARY '13		
Low-Capacitance Adaptor for DMMs	880	£8.55
★ 3-Input Stereo Audio Switcher		
– Main Board	881 } pair	£20.00
– Switch Board	882 } pair	
Stereo Compressor – Main Board	883	£12.63
Jump Start – iPod Speaker	884	£8.16
FEBRUARY '13		
10W LED Floodlight	885	£6.75
★ Crystal DAC (double-sided)	886	£18.46
Jump Start – Logic Probe	887	£6.42
MARCH '13		
Lightning Detector	888	£8.75
★ Digital Spirit Level	889	£8.75
★ SemTest – Part 2		
– Main/Lower Board	890	£16.52
– Display/Upper Board	891	£15.55
Interplanetary Voice	892	£8.75
Jump Start – DC Motor Controller	893	£8.55
APRIL '13		
Six-Decade Resistance Substitution Box	894	£10.10
SoftStarter (single-sided)	895	£8.36
Jump Start – Egg Timer	896	£8.36
SemTest HV DC Crowbar	897	£13.61
MAY '13		
Electronic Stethoscope	898	£9.72
PIC/AVR Programming Adaptor Board (d'ble-sided)	899	£23.33
Jump Start – Signal Injector Probe	900	£8.16

PROJECT TITLE	ORDER CODE	COST
JUNE '13		
USB Breakout Box	901	£7.97
Jump Start – Simple Radio Receiver	902	£8.94
Mix-It	903	£11.28
See NOTE above left regarding PCBs with eight digit codes		
JULY '13		
6-Decade Capacitance Substitution Box	04106121	£18.00
6-Decade Capacitance Substitution Box Panel/Lid	04106122	£18.00
SoftStarter For Power Tools	10107121	£9.00
High-Current Adaptor For Scopes And DMMs	04108121	£18.00
Jump Start – Temperature Alarm	904	£7.97
AUGUST '13		
Driveway Sentry	03107121	£18.00
Build A Vox	01207111	£22.00
Milliohm Meter Adaptor For DMMs	04102101	£18.00
SEPTEMBER '13		
MiniReg Adjustable Regulator	18112111	£6.00
USB Stereo Recording Interface	07106111	£18.00
Digital Sound Effects Module	09109121	£9.00
OCTOBER '13		
High-Temperature Thermometer/Thermostat	21105121	£22.00
LED Musicolour	16110121	£22.00

Please check price and availability in the latest issue.
A large number of older boards are listed on, and can be ordered from, our website.
Boards can only be supplied on a payment with order basis.

EPE SOFTWARE

★ All software programs for *EPE* Projects marked with a star, and others previously published can be downloaded free from the Library on our website, accessible via our home page at: **www.epemag.com**

PCB MASTERS

PCB masters for boards published from the March '06 issue onwards can also be downloaded from our website (**www.epemag.com**); **go to the 'Library' section.**

EPE PRINTED CIRCUIT BOARD SERVICE

Order Code **Project** **Quantity** **Price**

. . . .

Name . . . .

Address . . . .

. . . .

Tel. No. . . . .

I enclose payment of £ . . . . (cheque/PO in **£ sterling only**) to:

Everyday Practical Electronics

Card No. . . . .

Valid From . . . . Expiry Date . . . .

Card Security No. . . . . Maestro Issue No. . . . . . .

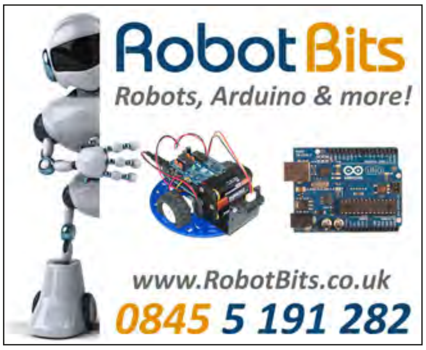
Signature . . . .

Note: You can also order PCBs by phone, Fax or Email or via the Shop on our website on a secure server:

http://www.epemag.com

CLASSIFIED
ADVERTISEMENTS

If you want your advertisements to be seen by the largest readership at the most economical price our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2·5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).
All cheques, postal orders, etc., to be made payable to Everyday Practical Electronics. **VAT must be added**. Advertisements, together with remittance, should be sent to Everyday Practical Electronics Advertisements, 113 Lynwood Drive, Merley, Wimborne, Dorset, BH21 1UU. Phone: 01202 880299. Fax: 01202 843233. Email: stewart.kearn@wimborne.co.uk. For rates and information on display and classified advertising please contact our Advertisement Manager, Stewart Kearn as above.



LOOKING FOR
AUDIO TRANSFORMERS
Visit Section 5
www.partridgeelectronics.co.uk

ELECTRONICS TECHNICIAN SERVICES
Trade qualified team with 30 years experience. Full workshop facility available. Can provide in-house assembly, and field installations in UK and overseas. Able to work in hostile environments (ie, Africa & Iraq).
Security specialist with extensive knowledge of Alarms, Cameras, Electric Fences, emergency power, and solar systems for remote locations. Highly experienced from PCB assembly to installations. Call today for further information
07729 614943 or
email dsalesmanager@gmail.com

BTEC ELECTRONICS TECHNICIAN TRAINING

NATIONAL ELECTRONICS
VCE ADVANCED ICT
HNC AND HND ELECTRONICS
FOUNDATION DEGREES
NVQ ENGINEERING AND IT
DESIGN AND TECHNOLOGY

LONDON ELECTRONICS COLLEGE
20 PENYWERN ROAD
EARLS COURT, LONDON SW5 9SU
TEL: (020) 7373 8721
www.lec.org.uk

BOWOOD ELECTRONICS LTD
Suppliers of Electronic Components
Place a secure order on our website or call our sales line
All major credit cards accepted
Web: www.bowood-electronics.co.uk
Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield, Derbyshire S40 2QR. Sales: 01246 200222
Send 60p stamp for catalogue

Like to advertise on the Classified page? Call Stewart Kearn on: 01202 880299

CANTERBURY WINDINGS
UK manufacturer of toroidal transformers (10VA to 3kVA)
All transformers made to order. No design fees. No minimum order.
www.canterburywindings.co.uk
01227 450810

SHERWOOD ELECTRONICS
CLOSING DOWN SALE
Send 69p Large letter stamp for list of available stock to: -
Sherwood Electronics
10 Newstead Street, Mansfield, Notts, NG19 6JJ

MISCELLANEOUS

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits. Visit **www.cstech.co.uk**

KITS, TOOLS, COMPONENTS. S.A.E. Catalogue. SIR-KIT ELECTRONICS, 52 Severn Road, Clacton, CO15 3RB, **http://sir-kit.webs.com**

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

ADVERTISERS INDEX

ADVERTISEMENT OFFICES:
113 LYNWOOD DRIVE, MERLEY, WIMBORNE,
DORSET BH21 1UU
PHONE: 01202 880299
FAX: 01202 843233
EMAIL: stewart.kearn@wimborne.co.uk
WEB: www.epemag.com
For editorial address and phone numbers see page 7

Next Month

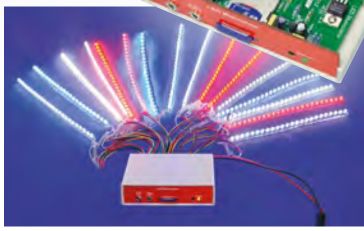
Content may be subject to change

Class-D amplifier

World's first DIY high-power, high-performance Class-D amplifier delivering 250W into 4Ω, and 150W into 8Ω. You've asked for it and now we are finally delivering it! This design has high power, extremely low harmonic distortion and is very quiet.

LED Musicolour: Light Up Your Music – Part 2

The new *LED Musicolour* consists of 16 strings of LEDs, which are individually controlled by 16 frequency bands to create a kaleidoscope of colour. It makes creating a spectacular light and music show easier than ever. In this second and final instalment, we explain how to build and test the unit and also detail how you can control it.



Timer For Fans And Lights

This simple circuit provides a turn-off delay for a 230V AC light or fan. It can be used to make a bathroom fan run for a set period after the switch has been turned off, or it can be used with a pushbutton to turn a light on for a specific time. The timer circuit consumes no standby power when the load is off.

Teach-In 2014: Raspberry Pi – Part 2

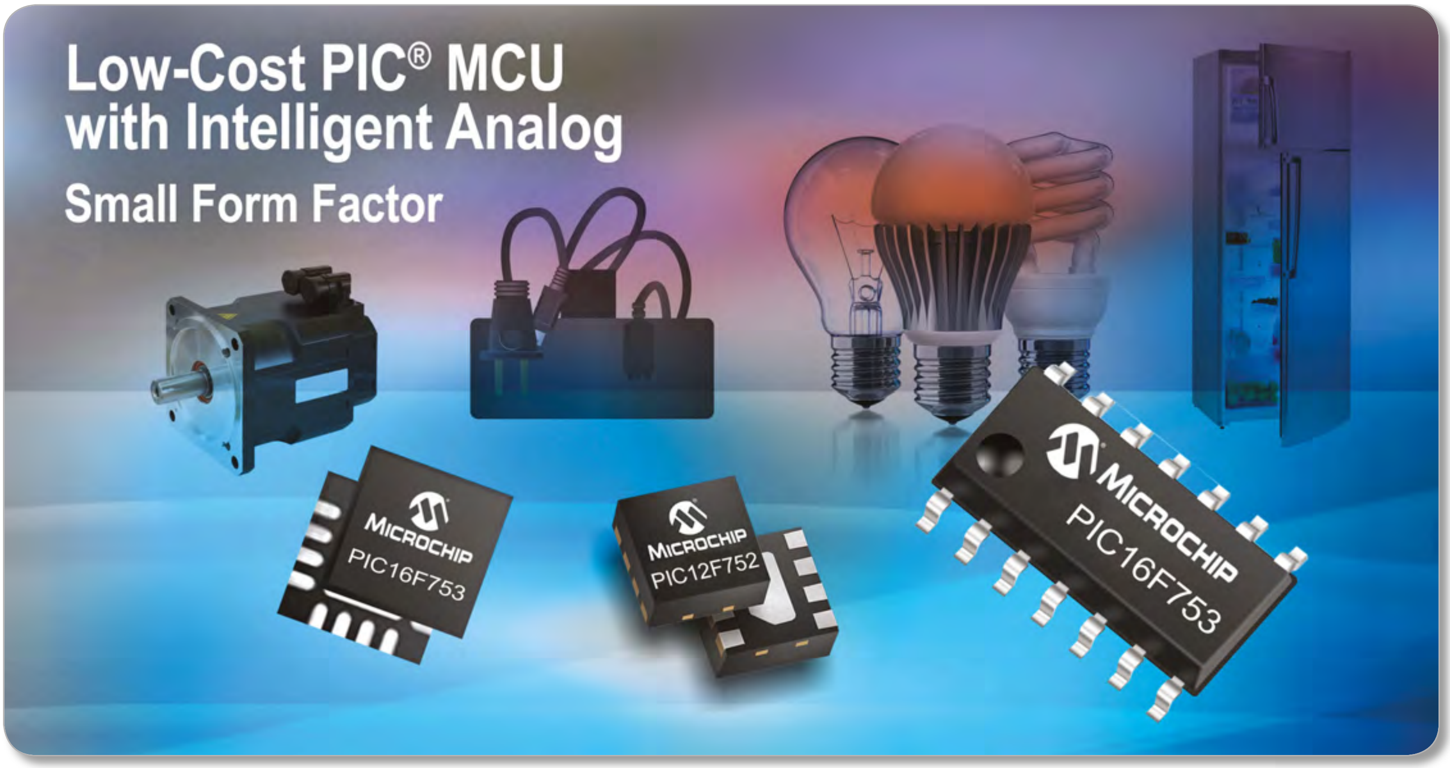
Next month, *Teach-In 2014* is all about connecting the Raspberry Pi to the real world. *Pi Class* will take you on a tour of the GUI and the applications that are bundled with the Raspberry Pi's operating system. *Pi Project* will explain how the GPIO interface works. Our feature for programmers, *Python Quickstart*, will provide you with an introduction to comparisons and loops and last, but by no means least, *Home Baking* will explain how to connect your Raspberry Pi to the Internet.

NOVEMBER '13 ISSUE ON SALE 3 OCTOBER 2013

Published on approximately the first Thursday of each month by Wimborne Publishing Ltd., 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU. Printed in England by Acorn Web Offset Ltd., Normanton, WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions INLAND: £21.95 (6 months); £41.50 (12 months); £78.00 (2 years). OVERSEAS: standard air service, £25.00 (6 months); £48.00 (12 months); £91.00 (2 years). Express airmail, £35.00 (6 months); £68.00 (12 months); £131.00 (2 years). Payments payable to "Everyday Practical Electronics", Subs Dept, Wimborne Publishing Ltd. Email: subs@epemag.wimborne.co.uk. EVERYDAY PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Does your design need analog intelligence from an 8-bit microcontroller?

PIC16F753 enables intelligent sensing plus efficient power conversion and management



Boost system performance and efficiency, while reducing system costs, with the enhanced analog capabilities of the PIC16F753 microcontroller. With its integrated Op Amp, slope compensation and range of core-independent peripherals, the PIC16F753 enables efficient power conversion and provides a versatile platform for adding intelligent sensing and power management to embedded designs.

Microchip's PIC16F753 8-bit microcontroller offers all the key features of the PIC12F752 and adds an Op Amp with 3 MHz of gain bandwidth product, in addition to slope compensation for switch-mode power supplies.

Non-overlapping, complementary waveforms, for comparator and PWM inputs, are provided by the on-chip complementary output generator, with dead-band, phase and blanking control, as well as auto shutdown and reset. The on-chip 8-channel, 10-bit ADC can be used to add sensing capabilities, including capacitive touch user interfaces, and two 50 mA outputs support direct driving of FETs. Also on-board are high-performance comparators, a 9-bit DAC, and a capture compare PWM module.

For high-voltage cost-sensitive designs, the PIC16HV753 features a shunt regulator for operation from 2V to a user-defined maximum voltage level, with less than 2 mA operation current.

GET STARTED TODAY!

- Free reference design for a high-power LED flashlight
- Choice of low-cost development tools:
  - PICDEM™ Lab Development Kit
  - PICkit™ 3
  - PICkit Low-Pin Count Demo Board

PICDEM™ Lab Development Kit (DM163045)

For more information, go to: www.microchip.com/get/eupic16f753

MICROCHIP

Microcontrollers • Digital Signal Controllers • Analog • Memory • Wireless